Small project - notes

Zanolin Lorenzo

February 23, 2024

Contents

Abstract

The main goal of this project is to dig into how Apple AirTags work, including their setup, security measures, and uses; since everything is tied to the FindMy network, we will also look into the security methods used in this network. The idea is to get a clear picture of how Apple AirTags are put together, how secure they are, and the different ways they're used in the FindMy network.

1 Introduction

In the rapidly evolving landscape of technological advancements, Apple's AirTags have emerged as a noteworthy innovation, offering a seamless solution to tracking and locating personal belongings. This paper delves into the architecture and security of AirTags, with a particular emphasis on the Ultra-Wideband (UWB) technology that underlies their functionality. Moreover, as AirTags operate within the framework of the FindMy network, this exploration extends to a thorough analysis of the security measures embedded in both the device and the overarching network; by unraveling the synergy between AirTags, UWB, and FindMy security, this paper aims to contribute insights into the robustness of the technological ecosystem that facilitates precise location tracking while addressing potential concerns surrounding privacy and data security.

In Section 2 we will explain some basics of UWB technology, continuing in Section 3 delving into the FindMy network analysis. Next we will deeply analyze the AirTag concept in 4, concluding with some future uses.

2 UWB

2.1 Overview

In the realm of wireless communication technologies, Ultra-Wideband (UWB) stands out as a revolutionary paradigm, redefining the capabilities and applications of wireless connectivity. UWB operates by employing extremely low-power, short-duration pulses that span a broad spectrum, allowing for high data transfer rates and precise positioning capabilities. Originally conceived for military and radar applications, UWB has gradually permeated diverse sectors, finding particular prominence in consumer electronics, healthcare, automotive systems, and the Internet of Things (IoT).

The unique feature of UWB is its capacity to send information across a large frequency range, usually several gigahertz; a UWB transmitter sends billions of radio pulses across the wide spectrum frequency and a UWB receiver then translates the pulses into data. The shorter the duration of the impulse, the more precise the distance measurement will be. UWB achieves real-time accuracy because as it sends up to 1 billion pulses per second. Because of its wide spectrum utilization, UWB can coexist peacefully with other wireless technologies and send out massive volumes of data quickly. Since its ability to transmit info

across a wide radio bandwidth, from 500MHz to several gigahertz, this technology has a short range of operation. According to [5], due to the low energy density and the pseudo-random (PR) characteristics of the transmitted signal, the UWB signal is noise-like which makes unintended detection difficult. By sending pulses in patterns, UWB encodes information and it takes between 32 and 128 pulses to encode a single bit of data, but given how fast the bits arrive, that enables data rates of 7 to 27 megabits per second. To increase UWB's range and reception reliability, a *MIMO* (multiple-input and multiple-output), distributed antenna system has been added to the standard that enables short-range networks. The antennas can be embedded into a smartphone or other devices such as a wristband or smart key.

According to [2], Apple-designed U1 chip uses Ultra Wideband technology for spatial awareness— allowing iPhone 11, iPhone 11 Pro, and iPhone 11 Pro Max or later iPhone models to precisely locate other U1-equipped Apple devices. When two U1 devices come close to eachother, the two start measuring their exact distance. The ranging is accomplished through Time of Flight (ToF), which is the time it takes for a pulse to get from point A to point B. According to IEEE 802.15.4a [10], UWB can determine the relative position of other devices in the line of sight even up to 200 meters.

A difference w.r.t. Bluetooth Low Energy is that with Bluetooth you can't really measure location or distance. What you can do is to detect if a device like a smartphone is within a range of another device. Ultra-Wideband, in comparison, provides a much higher accuracy (up to a few centimeters). In contrast to Bluetooth Low Energy, the distance it measures is not based on the signal strength, but the time it takes the signal to travel from point A (smartphone) to point B (UWB tag). The following table represent the principal differences between the two technologies:

	UWB	**Bluetooth (BLE Beacons)**
Battery	Low consumption	Low consumption
Range	up to 200 meters (656 feet)	up to 70 meters (230 feet)
Accuracy	10 centimeters (3.9 inches)	up to a meter
Cost	Low	Low

2.2 Security analysis

The fact that UWB pulses are resistant to the multipath effect[1] is one of its key characteristics. This occurs when radio waves are reflected or refracted by artificial or natural objects near to the primary signal channel, causing the signal to reach the receiver via many paths. Positioning accuracy is improved by immunity to the multipath effect, particularly when compared to other technologies that are more vulnerable. Moreover, UWB's resilience to jamming and

[1] Multipath interference occurs when a signal from a transmitter arrives at a receiver via two or more routes; typically there is a direct path plus a number of indirect paths caused by reflections

narrowband fading makes it a particularly reliable technology choice, even when several UWB systems are being used at once.

Another important aspect to consider is the resistance from Relay Attacks, which is a vulnerability of the majority of signals-based architectures. Within this attacks, the goal is to trick a car into thinking the key and owner are close by using two people with hacking devices. The first relays signals from the car to the second thief, who transmits the signal to the house. The key responds, allowing entry into the car. The relay attack intercepts and amplifies wireless signals used to unlock the door and start the car, despite the key's distance. With UWB that would not be possible due to its working scheme: to ensure the car doesn't have to make assumptions, rapid measurements are utilized to establish distance very precisely. Any attempt to relay attack or intercept the UWB signal will merely cause the answering device's acknowledgement signals to arrive later, indicating to the UWB-based lock and ignition that the responding device is actually farther away rather than closer. The car's presumption is replaced with assurance when using UWB, which greatly increases the security of the passive keyless entry system.

The implemented security measures, taken by Apple, are the following: MAC address randomization and frame sequence number randomization [2].

MAC address randomization

Apple platforms use a randomized media access control address (MAC address) when performing scans when not associated with another device. Because a device's MAC address changes when disconnected from another device, it can't be used to persistently track a device by passive observers of traffic.

Frame sequence number randomization

Apple devices randomize the sequence numbers whenever a MAC address is changed to a new randomized address.

3 FindMy network

3.1 Overview

Apple created the sophisticated and extensive location-tracking technology, known as the FindMy network, which is intended to assist consumers in finding their compatible third-party accessories, such as Apple AirTags, as well as their Apple devices, including iPhones, iPads, Macs, and AirPods. By utilizing a blend of Ultra-Wideband (UWB) and Bluetooth Low Energy (BLE) technologies, the FindMy network facilitates accurate and instantaneous location tracking.

Because it may be used to find everything you need to find, Apple combined the Find My Friends and Find My iPhone apps into a single app that is simply called *Find My*. Subsequently, Apple has consistently enhanced the

Find My app, incorporating functionalities such as monitoring when a iPhone is disconnected, when it is turned off, and when it has been wiped.

The app is organized into several sections, accessible by tapping the tabs at the bottom. On the left, you can find people, in the middle, you can find your own devices and items, such as AirTags, and Find My-enabled Bluetooth items; finally on the right there's a *Me* tab with all of your settings and info, as we can see in Figure 1.

There are several useful features within this app, as example:

- *Separation alerts*: Designed to let you know if you leave an Apple device, a device attached to an AirTag, or a Find My-enabled third-party device behind.
- *Locating friends and sharing location*: Implemented to locate friends and family members that have shared their location with you. You can view their location using the *People* tab within the Find My app.
- *Locating devices without connection*: Lets your lost devices be located even when not connected to WiFi or LTE by leveraging Bluetooth BLE and proximity to other nearby Apple devices. When your lost device is offline but close to another device, it's able to connect to that other device over Bluetooth and relay its location. We will come back on this...
- *Anti-stalking measures*: Designed to let you know if there's a Bluetooth item near you. You will receive a notification when an unknown item is found and moving with you, so you can make sure no one slips an AirTag or other Find My Bluetooth device into your things to track you. We will also come back on this following the studies done in [7].
- *Precision finding*: Takes advantage of the U1 chip within AirTags and the iPhone 13, iPhone 12 and iPhone 11 models. Apple's U1 chip uses ultra-wide band techology to precisely locate and communicate with other U1-equipped devices, enabling AirTags and iPhone 13/12/11 models to work together.

3.2 Functioning

As for the functioning, we will use the content of [12, 9, 4, 14] to describe the entire process. First of all we want to register our products within the application; in case of a non-accessory item, it will automatically be inserted in the *Devices* section. In case of accessories, like AirTag, you need to manually add them using the *Items* section.

Now you should be able to localize online devices from the map; in case of a close AirTag you can also use the *Precision finding* to locate the object using U1 chip (this function is only available for iPhone 11 and aboves), as represented in Figure 2. The technology fuses input from the iPhone's camera,

Figure 1: FindMy screenshots.

ARKit, accelerometer and gyroscope in order to guide the user to their AirTag using a combination of sound, haptics, and visual feedback.

In case of a stolen device, you can always activate *Lost Mode*; it will send you a notification as soon as the iphone is relevated, which means that if the phone will be connected to a network or via BLE to other Apple devices the owner will receive a message from FindMy containing the position of the device. Also, in this modality, all the registered cards in Apple Pay will be temporaneously disabled.

This technology seems already pretty useful, but if a device is powered off how can it be localized? With the introduction of iOS 15 and the U1 chips, Apple claimed that even a U1-based iPhone can be localized even when turned off if and only if the Bluetooth is turned on. From Apple: With iOS 15, your iPhone is still traceable through the Find My network even when the device is powered off. It seems that with iOS 15, the phone is not really fully powered off, it stays in a low-power state and acts like an AirTag, allowing any nearby iOS device to pick up the Bluetooth signal and send back its location. This also means if the iPhone runs out of battery during the day, the owner still has a chance of finding its location for several more hours. In fact, Apple says the location tracking will even keep working whilst the phone is reset to factory settings with Activation Lock enabled. And what about other devices, like Macbooks? Since Apple has not commented on the matter, the author conducted some tests and discovered that, regrettably, even machines with high specifications lack the U1 chip and are therefore not localizable when turned off.

As already written, AirTags can be tracked through the *Items* tab and have all of the tracking features available to Apple devices like iPhones and iPads.

Figure 2: Precision Finding.

There's a Lost Mode, and AirTags can also take advantage of the Find My network that allows them to be tracked by billions of iPhones, iPads, and Macs when they're out of range of the owner devices. The only way to stop an AirTag from being fully detectable is to remove the CR2032 coin battery (or when it fully discharges).

3.3 Security analysis

In this section we will analyze the security measures implemented in FindMy to increase the safety of the architecture. Let us add some more details in the process of localizing an object. Let us first define a few terms in accordance with [8]:

- *Owner Devices*: Owner devices share a common Apple ID and can use the Find My application on macOS, iOS and ipadOS to search for any devices of the same owner.
- *Lost Devices*: Devices that determine to be in a lost state start sending out BLE advertisements with a public key to be discovered by finder devices. Apple devices are considered to be lost when they lose Internet connectivity. Third-party accessories [15] are small battery-powered devices that can be attached to a personal item and are set up through an owner device. Accessories determine to be lost when they lose their BLE connection to the owner device.
- *Finder Devices*: Finder devices form the core of the FindMy network. As of 2023, only iPhones and iPads with a GPS module are offering finder

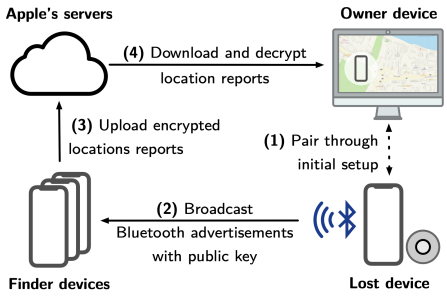

Figure 3: FindMy workflow (simplified).

capabilities. Finder devices can discover lost devices and accessories by scanning for BLE advertisements. Upon receiving an advertisement, a finder creates an end-to-end encrypted location report that includes its current location and sends it to Apple's servers. We will see what are the contents of the report.

- *Apple's Servers*: Apple's servers store the location reports submitted by finder devices. Only owner devices can fetch those reports and decrypt them locally.

An online device can simply report its location to the user via iCloud. Find My works offline by sending out short range Bluetooth signals from the missing device that can be detected by other Apple devices in use nearby. Those nearby devices then relay the detected location of the missing device to iCloud so users can locate it in the Find My app while protecting the privacy and security of all the users involved. An example of the previous workflow is represented in figure 3.

All finder devices regularly scan for FindMy advertisements. The finder creates and uploads an encrypted position report to Apple's servers whenever it receives a packet in the FindMy advertising format; finder devices are likely to use less energy and bandwidth because they gather reports over time and transfer them in batches on a regular basis. The evaluation done in [8] discovered that the median time from generating to uploading a location report is 26 min and the delay can increase to several hours if the finder device is in a low power mode.

We will now delve more in the used Cryptography protocols.

3.3.1 Advertisement keys

FindMy utilises Elliptic Curves [13]; according to [2, 8] the process to generate *advertisement keys* is the following:

1. Each owner device generates a private/public key pair (d_0, p_0) on the NIST $P-224$ curve and a 32-byte symmetric key SK_0 that together form the

master beacon key. Those keys are never sent out via BLE and are used to derive the rolling advertisement keys included in the BLE advertisements. Note that device tracking is hard since rolling keys can be deterministically derived if and only if one knows the initial input keys (d_0, p_0) and $SK0$.

2. Advertisement keys (d_i, p_i) are iteratively calculated as follows using the ANSI X.963 key derivation function KDF [1] with $SHA-256$ [6] and a generator G of the NIST P-224 curve:

$$SK_i = KDF(SK_{i-1},\ 'update',\ 32) \tag{1}$$
$$(u_i, v_i) = KDF(SK_i,\ 'diversify',\ 72) \tag{2}$$
$$d_i = (d_0 * u_i) + v_i \tag{3}$$
$$p_i = d_i * G \tag{4}$$

where Equation (1) derives a new symmetric key from the last used symmetric key with 32 bytes length. Equation (2) derives the so-called "anti-tracking" keys u_i and v_i from the new symmetric key with a length of 36 bytes each. Finally, Eqs. (3) and (4) create the advertisement key pair via EC point multiplication using the anti-tracking keys and the master beacon key d_0.

3.3.2 Key synchronization

To download and decode location information, the advertisement keys must be accessed by all owner devices. In order to synchronize the master beacon keys, FindMy uses iCloud to encrypt a property list file in the Galois/Counter mode of the Advanced Encryption Standard (AES-GCM) [11]. The file's decryption key is kept in the iCloud keychain underneath the label "Beacon Store".

3.3.3 Encryption

The BLE advertisements sent out by a lost device contain an EC public key p_i. A finder device that receives such an advertisement determines its current location and encrypts the location with p_i. OF employs the Elliptic Curve Integrated Encryption Scheme (ECIES), which performs an ephemeral Elliptic Curve Diffie-Hellmann (ECDH) key exchange to derive a shared secret used to encrypt the report. According to [8], the finder's encryption algorithm works as follows:

1. Generate a new ephemeral key (d', p') on the NIST P-224 curve for a received FindMy lost advertisement.

2. Perform ECDH using the ephemeral private key d' and the advertised public key p_i to generate a shared secret.

3. Derive a symmetric key with ANSI X.963 KDF on the shared secret with the advertised public key as entropy and SHA-256 as the hash function.

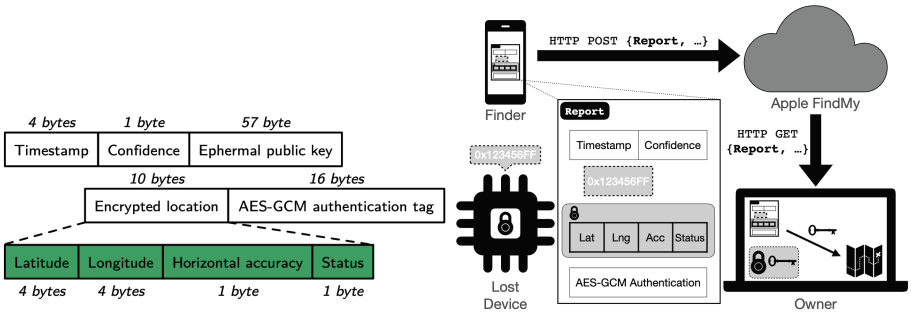

Figure 4: (a) Binary format of a location report (b) Example of sent payload.

4. Use the first 16 bytes as the encryption key e'.
5. Use the last 16 bytes as an initialization vector IV.
6. Encrypt the location report under e' and the IV with AES-GCM.

The ephemeral public key p' and the authentication tag of AES-GCM are part of the uploaded message All location reports are identified by an ID which is the SHA-256 hash of p_i, as we can see in figures 4.

3.3.4 Decryption

An owner device that retrieves encrypted location reports follows the inverse of the encryption procedure:

1. The owner device selects the proper advertisement keys (d_i, p_i) based on the hashed p_i of the location report.
2. It performs the ECDH key exchange with the finder's ephemeral public key p' and the lost device's private key d_i to compute the symmetric key e' and the IV.
3. Now, the owner can use e' and IV to decrypt the location report.

4 AirTag

4.1 Overview

An AirTag is a small tracking device developed and sold by Apple; it is intended to assist users in finding and monitoring objects that are easily misplaced or lost, such as wallets, bags, and keys. By connecting to compatible Apple devices (iPhones, iPads, and Macs) through Bluetooth, the AirTag enables users to locate the linked object and, subsequently, the AirTag itself using the Find My app. In order to help with lost item recovery, the gadget has capabilities

including accurate location tracking, proximity notifications, and a Lost Mode. Its design prioritizes privacy and security, safeguarding user data through the use of encryption and anonymization. Support for the AirTag was introduced in iOS 14.5 and Apple lists as compatible all the devices that support iOS 14 and iPadOS.

4.2 Architecture

First of all we will analyze the architecture of this product; most of the presented stuff has been reverse engineered by Adam Catley in [3]. We will cover both the hardware and the software aspects of this kind of devices.

4.2.1 Software analysis

As for the software part, we can identify various states in which AirTag can be; we will now quickly present all of them.

- *Not registered*: When the AirTag is brand new, has been reset, or has been removed from the FindMy network. Waits to be connected to while advertising itself every 33ms.
- *Initialisation*: The AirTag is being registered to an Apple ID and a public/private key pair is generated and shared between the AirTag and the connected iOS device.
- *Connected*: The owner's device is in range. No broadcasts occur.
- *Disconnected*: The owner's device is out of range. Broadcasts identity every 2000ms.
- *Out of sync*: Happens when an AirTag reboots while separated from its owner's device. Acts like Disconnected but absolute time is lost so events are relative to time since power-up. Identity resets to initial value.
- *Lost*: Occurs 3 days after Disconnected or Out of sync begin. Moves to Waiting for motion every 6 hours.
- *Waiting for motion*: Samples the accelerometer every 10 seconds until motion is detected.
- *Sound alert*: A command to play a noise is received from either a connected device or by detecting motion. Lasts a maximum of 20 seconds.
- *Precision finding*: Triggered by the owner's device while in Connected. Is overridden by Sound alert.

As we will see in section 4.2.2, the AirTag uses a NFC antenna to allow anyone who finds an AirTag to potentially identify the owner, even if they have an Android device. The tag can only be read when the AirTag is powered by a battery and it contains a URL to uniquely identify the AirTag, depending on

its current state. There are some parameters that are attached to the URL, used to identify the device; we will briefly present them. Keep in mind that all of the following parameters are fixed for a single unit, even after power cycles, long runtime, resets and modes.

- *pid*: Product ID for AirTag.
- *b*: Something related to the battery.
- *pt*: UWB Precision Tracking version.
- *fv*: Firmware version.
- *dg*: Something related to diagnostic.
- *z*: Unknown.
- *bt*: Bluetooth address; will be present only when Unregistered.
- *st*: Serial number of the tag (the one printed on the device under the battery); will be present only when Unregistered.
- *bp*: Bluetooth protocol version; will be present only when Unregistered.

In the **Unregistered** state, the stored URL has the following format: `https://found.apple.com/airtag?pid=5500&b=00&pt=004c&fv=00100e10&dg=00&z=00&bt=A0B1C2D3E4F5&sr=ABCDEF123456&bp=0015`

Once the device is **registered**, the parameters *bt, sr, bp* will be removed and replaced with a single anonymous identifier *pi*, which is the only parameter that changes at least every 15 minutes when the Bluetooth address and/or the advertising data changes. It is likely the current P-224 public key p_i presented in 3.3.1 or the SHA-224 of it.

4.2.2 Hardware analysis

4.3 Security Analysis

4.3.1 Anti-stalking measurements

4.4 Future applications

References

[1] en. URL: https://csrc.nist.gov/CSRC/media/Events/Key-Management-Workshop-2000/documents/x963_overview.pdf.

[2] Apple. *Apple Platform Security*. en. May 2022. URL: https://support.apple.com/en-gb/guide/security/welcome/web.

[3] Adam Catley. *Apple AirTag Reverse Engineering - Adam Catley*. en. URL: https://adamcatley.com/AirTag.html#pcb-overview.

[4] Juli Clover. "Find my App: Everything to Know". en-US. In: *MacRumors* (Feb. 2022). URL: https://www.macrumors.com/guide/find-my/.

[5] Maria-Gabriella Di Benedetto. "UWB communication systems: a comprehensive overview". In: (2006).

[6] Henri Gilbert and Helena Handschuh. "Security analysis of SHA-256 and sisters". In: *International workshop on selected areas in cryptography*. Springer. 2003, pp. 175–193.

[7] Alexander Heinrich, Niklas Bittner, and Matthias Hollick. "AirGuard-protecting android users from stalking attacks by apple find my devices". In: *Proceedings of the 15th ACM Conference on Security and Privacy in Wireless and Mobile Networks*. 2022, pp. 26–38.

[8] Alexander Heinrich et al. "Who Can Find My Devices? Security and Privacy of Apple's Crowd-Sourced Bluetooth Location Tracking System". In: *CoRR* abs/2103.02282 (2021). arXiv: 2103.02282. URL: https://arxiv.org/abs/2103.02282.

[9] Mahmoud Itani. *How are iPhones still findable even when turned off*. en. Oct. 2021. URL: https://www.xda-developers.com/iphone-findable-turned-off/#:~:text=Most%20time%2C%20U1%20is%20on,the%20connected%20device%20via%20Bluetooth..

[10] Eirini Karapistoli et al. "An overview of the IEEE 802.15.4a Standard". In: *IEEE Communications Magazine* 48.1 (2010), pp. 47–53. DOI: 10.1109/MCOM.2010.5394030.

[11] Emilia Käsper and Peter Schwabe. "Faster and timing-attack resistant AES-GCM". In: *International Workshop on Cryptographic Hardware and Embedded Systems*. Springer. 2009, pp. 1–17.

[12] Benjamin Mayo. *iOS 15: Find My network can still find your iPhone when it is powered off, or factory reset - 9to5Mac*. en-US. June 2021. URL: https://9to5mac.com/2021/06/07/ios-15-find-my-network-can-find-your-iphone-when-it-is-powered-off/.

[13] Victor S Miller. "Use of elliptic curves in cryptography". In: *Conference on the theory and application of cryptographic techniques*. Springer. 1985, pp. 417–426.

[14] Britta O'Boyle. *What is Precision Finding on AirTags and how does it work?* en. Nov. 2021. URL: https://www.pocket-lint.com/phones/news/apple/156605-what-is-precision-finding-on-apple-airtags-and-how-does-it-work/.

[15] Apple Inc. All rights reserved. *MFi Program - create accessories that communicate with Apple devices using MFi technologies and components*. en. 2023. URL: https://mfi.apple.com/.