Bluetooth (/tags/#Bluetooth)

Annoying Apple Fans: The Flipper Zero Bluetooth Prank Revealed

Posted by Tech on September 1, 2023

Annoying Apple Fans: The Flipper Zero Bluetooth Prank Revealed

The Bluetooth Low Energy (BLE) protocol, a cornerstone of modern wireless communication, has been instrumental in enabling seamless interactions between devices. Its advertising packets, in particular, are broadcast signals that devices use to announce their presence and capabilities. Apple's ecosystem, with its myriad of interconnected devices, heavily relies on these packets for functionalities ranging from AirDrop file transfers to Apple Watch connectivity.

In November 2022, I released a YouTube video discussing AirTag spoofing. We're going to extend that to other services.

(http://www.youtube.com/watch?v=m_-nMw5bzjI)

Bluetooth Low Energy (BLE) and ADV Packets:

Bluetooth Low Energy (BLE), as part of the Bluetooth 4.0 specification, was introduced to cater to applications that require minimal power consumption. It's especially suitable for devices that need periodic or occasional transfer of data.

One of the primary mechanisms by which BLE devices communicate or make their presence known is through advertising packets, commonly referred to as ADV packets.

ADV Packets:

> ***Purpose:*** *ADV packets are broadcasted by devices to announce their presence. These can be picked up by any device that's listening, without pairing.*

Types of Advertising:

- **Connectable Directed Advertising:** Targeted advertising for a specific device.
- **Non-connectable Undirected Advertising:** For devices broadcasting information without connecting, like a beacon.
- **Scannable Undirected Advertising:** Allows a scan request from a receiving device but doesn't establish a connection.

Data Payload: Contains information like the device's name or services it offers.

Frequency: Devices can adjust their ADV packet broadcast frequency, balancing power consumption against discoverability.

Implications for Apple

Apple's ecosystem relies on BLE for many integrations and features. Here's the role of ADV packets for Apple:

- **AirDrop:** Uses BLE to discover nearby devices with ADV packets identifying potential share targets.
- **Handoff:** Uses BLE to detect nearby devices, with ADV packets

determining device presence.

- **Apple Watch:** Uses BLE for a low-energy connection with the iPhone, with ADV packets aiding in discovery and connection.
- **HomeKit:** Many HomeKit devices use BLE, advertising their status or availability.
- **Security and Privacy:** Apple ensures user privacy by implementing measures like rotating the Bluetooth address to prevent tracking, even though ADV packets can be picked up by any listening device.
- **iBeacons:** Uses BLE's non-connectable undirected advertising to broadcast a unique identifier for location-based services.

In conclusion, ADV packets in BLE are crucial for many of Apple's features. Apple harnesses BLE advertising for efficient device communication while prioritizing user privacy and security.

Flipper Zero Bluetooth Prank Revealed

Enter the Flipper Zero, a multi-tool device for hackers and tinkerers. One of its capabilities is to interact with BLE protocols, and more specifically, to mimic or spoof these advertising packets.

When a device like Flipper Zero mimics the advertising packets of legitimate devices or services, it can create a plethora of phantom devices in the vicinity of an iOS user. Imagine searching for a device to connect to and being presented with a list of dozens, if not hundreds, of fake device names. Or attempting an AirDrop and being flooded with fictitious recipients. It's not just a minor inconvenience; it can disrupt the seamless experience that

Apple users are accustomed to.

But why would someone do this? The reasons can vary:

- **Pranks:** Some might find it amusing to watch Apple aficionados grapple with a sudden influx of mysterious devices appearing on their screens.
- **Testing and Research:** Cybersecurity professionals might use such tactics to study vulnerabilities or test the robustness of BLE implementations.
- **Malicious Intent:** While less common, there's potential for malicious actors to exploit this for nefarious purposes, such as a type of phishing attack by mimicking trusted notifications. (Blog Post being written)

For iOS users, this mimicry can be more than just an annoyance. It can lead to confusion, disrupt workflows, and in rare cases, pose security concerns. It underscores the importance of being aware of the devices around us and the potential vulnerabilities inherent in wireless communications.

But why shed light on this? It's essential to note that the Flipper Zero's range is limited. To truly impact an iOS device, you'd need to be in close proximity. This isn't about widespread disruption, but rather an exploration of the playful potential and boundaries of wireless communication.

Here's how it's done.

We'll be updating the Flipper Zero by altering a specific file responsible for its Bluetooth functionality, then compiling and applying the firmware update.

Make sure you have enough space and clone the source code:

> *git clone –recursive https://github.com/flipperdevices/flipperzero-firmware.git*

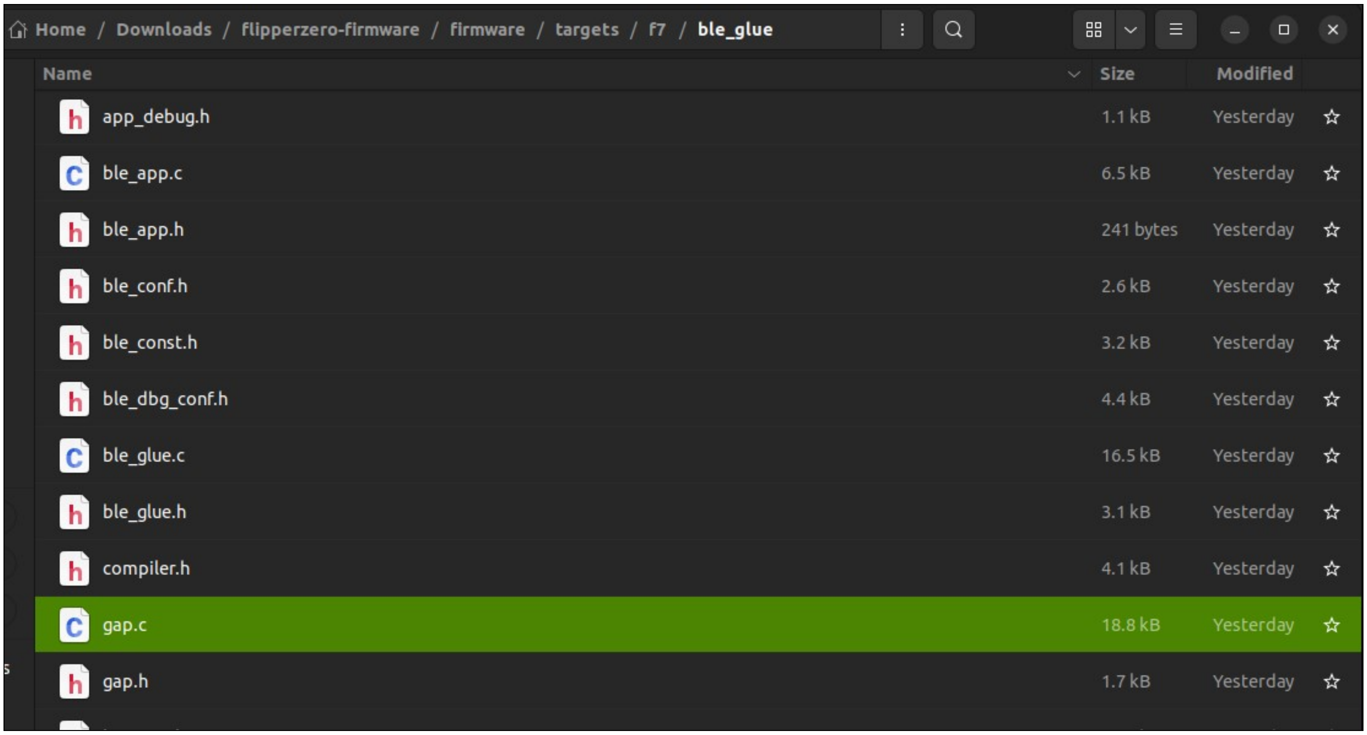
I’ve streamlined the process for you, allowing you to effortlessly select the type of notification you’d like to display on nearby iOS devices.

Select Device Type ⌄

Copy Code

Updating the ‘gap.c’ file and then compiling the updated firmware.

After your selection of notification type, copy the provided code output and then update the contents within the ‘gap.c’ file located at the specified location.

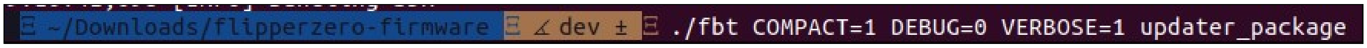

Home / Downloads / flipperzero-firmware / firmware / targets / f7 / ble_glue

Name	Size	Modified
app_debug.h	1.1 kB	Yesterday
ble_app.c	6.5 kB	Yesterday
ble_app.h	241 bytes	Yesterday
ble_conf.h	2.6 kB	Yesterday
ble_const.h	3.2 kB	Yesterday
ble_dbg_conf.h	4.4 kB	Yesterday
ble_glue.c	16.5 kB	Yesterday
ble_glue.h	3.1 kB	Yesterday
compiler.h	4.1 kB	Yesterday
gap.c	18.8 kB	Yesterday
gap.h	1.7 kB	Yesterday

After updating the file, proceed to compile the firmware. Ensure you're in the root directory before executing the following command:

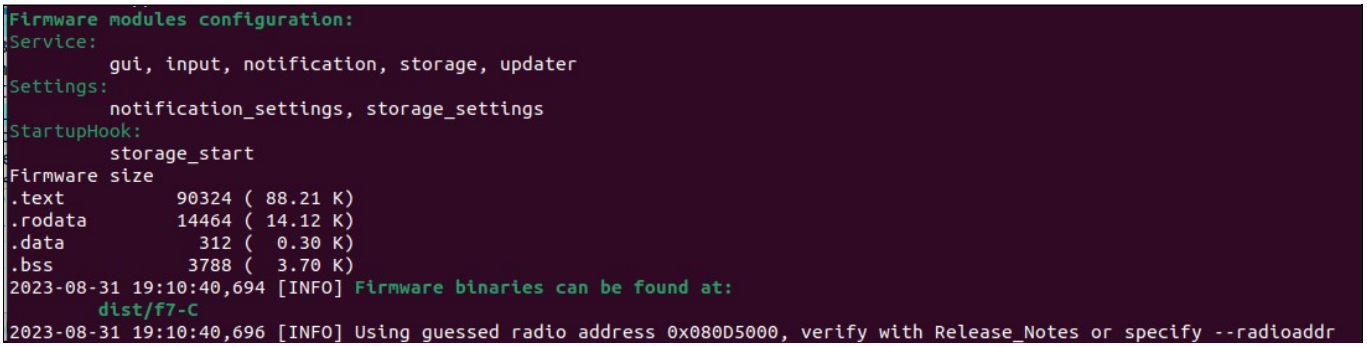
> *./fbt COMPACT=1 DEBUG=0 VERBOSE=1 updater_package*

As an illustration:

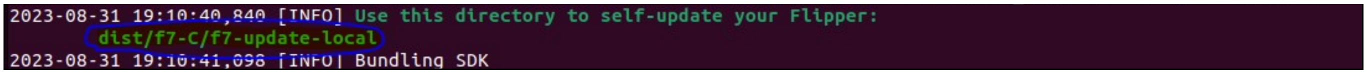


```
~/Downloads/flipperzero-firmware  dev ±  ./fbt COMPACT=1 DEBUG=0 VERBOSE=1 updater_package
```

Once the command executes, near the end, you'll identify the directory where the compilation occurred:

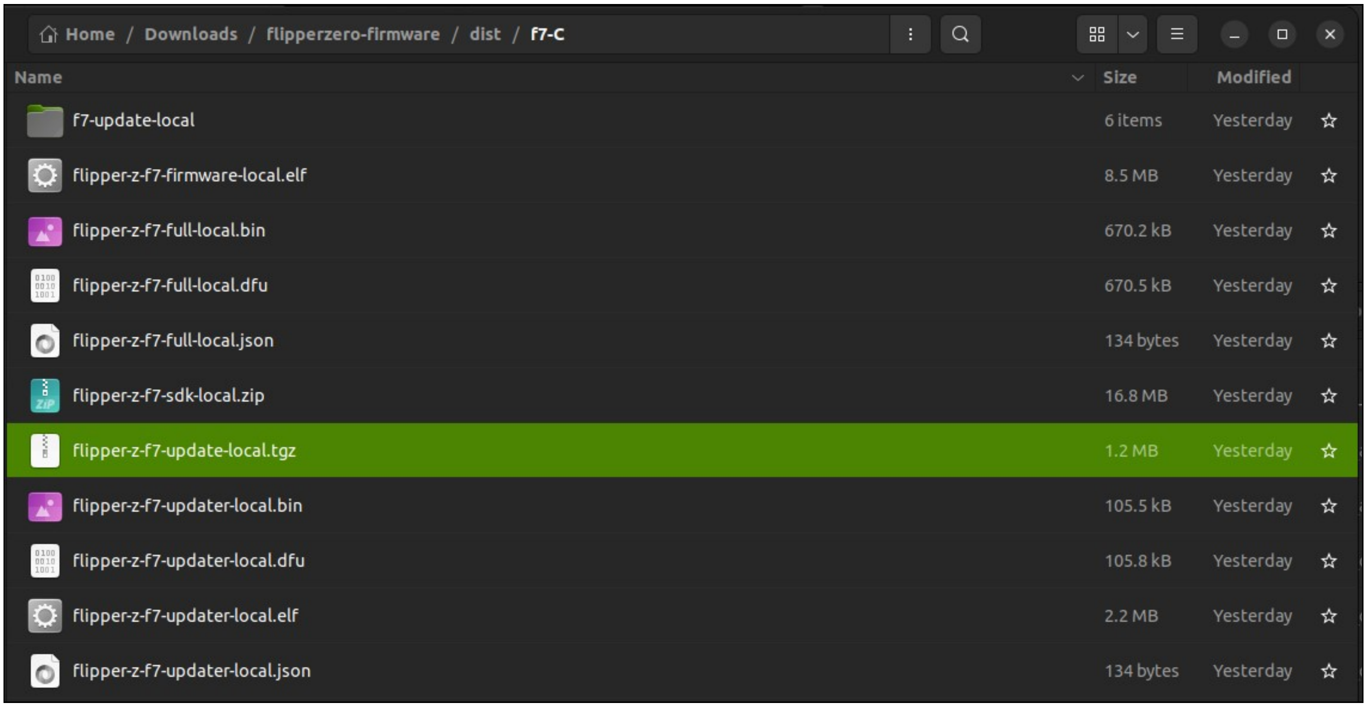


```
Firmware modules configuration:
Service:
         gui, input, notification, storage, updater
Settings:
         notification_settings, storage_settings
StartupHook:
         storage_start
Firmware size
.text          90324 ( 88.21 K)
.rodata        14464 ( 14.12 K)
.data            312 (  0.30 K)
.bss            3788 (  3.70 K)
2023-08-31 19:10:40,694 [INFO] Firmware binaries can be found at:
        dist/f7-C
2023-08-31 19:10:40,696 [INFO] Using guessed radio address 0x080D5000, verify with Release_Notes or specify --radioaddr
```



```
2023-08-31 19:10:40,840 [INFO] Use this directory to self-update your Flipper:
        dist/f7-C/f7-update-local
2023-08-31 19:10:41,098 [INFO] Bundling SDK
```

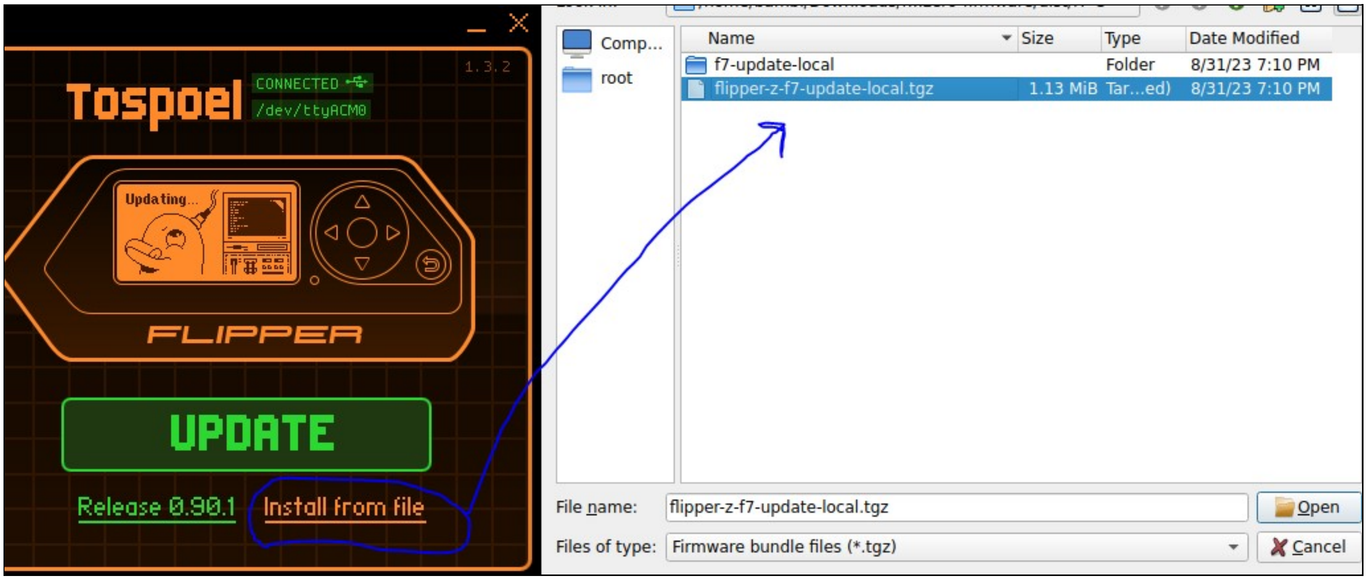
Within that folder, your new compiled firmware is the .tgz file that we will open in the qFlipper app.



Home / Downloads / flipperzero-firmware / dist / f7-C

Name	Size	Modified
f7-update-local	6 items	Yesterday
flipper-z-f7-firmware-local.elf	8.5 MB	Yesterday
flipper-z-f7-full-local.bin	670.2 kB	Yesterday
flipper-z-f7-full-local.dfu	670.5 kB	Yesterday
flipper-z-f7-full-local.json	134 bytes	Yesterday
flipper-z-f7-sdk-local.zip	16.8 MB	Yesterday
flipper-z-f7-update-local.tgz	1.2 MB	Yesterday
flipper-z-f7-updater-local.bin	105.5 kB	Yesterday
flipper-z-f7-updater-local.dfu	105.8 kB	Yesterday
flipper-z-f7-updater-local.elf	2.2 MB	Yesterday
flipper-z-f7-updater-local.json	134 bytes	Yesterday

In that directory, the newly compiled firmware is represented by the .tgz file, which we'll load into the qFlipper app.



Name	Size	Type	Date Modified
f7-update-local		Folder	8/31/23 7:10 PM
flipper-z-f7-update-local.tgz	1.13 MiB	Tar...ed)	8/31/23 7:10 PM

File name: flipper-z-f7-update-local.tgz

Files of type: Firmware bundle files (*.tgz)

Have fun!

Here are some examples of what the notifications look like.

← PREVIOUS POST (/2023/03/11/TILE-ANTI-THEFT-MODE-SEPARATING-HYPE-FROM-REALITY/)

FEATURED TAGS (/tags/)

Reverse Engineering (/tags/#Reverse Engineering) Programming (/tags/#Programming)
Hardware (/tags/#Hardware) Exploit Development (/tags/#Exploit Development) Pentesting (/tags/#Pentesting)
Shellcoding (/tags/#Shellcoding) x86 (/tags/#x86) Linux (/tags/#Linux) RedTeam (/tags/#RedTeam)
Bluetooth (/tags/#Bluetooth)

● (https://twitter.com/tech) ● (https://github.com/techryptic)