Pwning Samsung: An Attempt at Hardware and Bluetooth Hacking

Jose Vasquez, Ali Chowdhury, and Andrew Schwartz

Agenda

1. Introduction and Motivation
2. Background
3. Related Works
4. Threat Model
5. Hardware Analysis
6. Bluetooth Analysis
7. Conclusion

1.

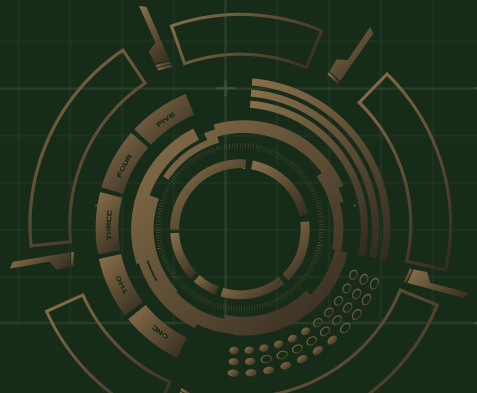
Introduction & Motivation

Samsung SmartTag 2

Overview:

Samsung's equivalent to Apple Airtags. Essentially a Bluetooth tracker that use Samsung's Smart Things and Find My Mobile (FMM) Networks.

Commonly used for tracking keys, pets, and luggage.



Why SmartTag 2?

- Positioned as a significant competitor to Apple AirTags
- Enhanced tracking capabilities leveraging BLE and UWB technologies
- Promises secure tracking of personal items through Samsung's ecosystem
- Unlike AirTags, SmartTag 2 remains largely unexplored, with little to no prior research or documented vulnerabilities

2.

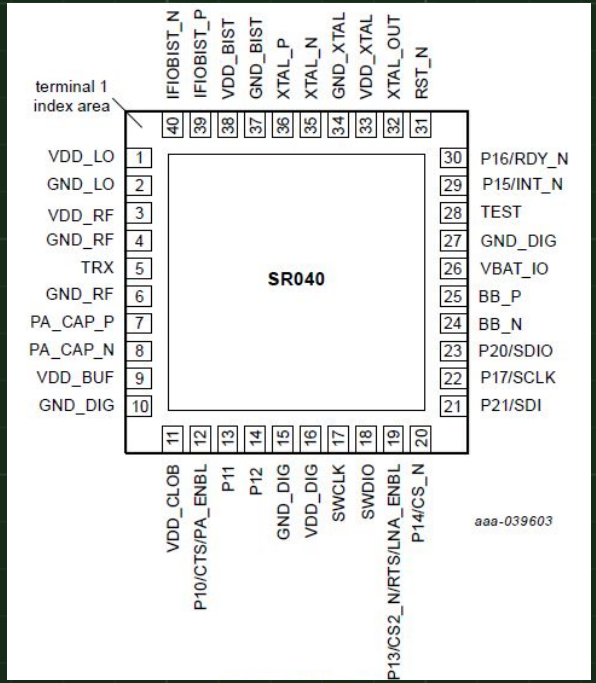
Background

Ultra-Wideband (UWB) Chip

Enables high-accuracy location tracking in IoT devices.

Key Features:

- **Precision Ranging:** Achieves ±10 cm accuracy, even in non-line-of-sight conditions.
- **Low Power Operation:** Designed for coin cell batteries, with power-saving mechanisms.
- **Security:** Implements Scrambled Timestamp Sequences (STS), compliant with NIST SP 800-90A for secure data exchange.
- **Integrated Processor:** ARM Cortex-M33 with TrustZone for secure firmware updates and processing.



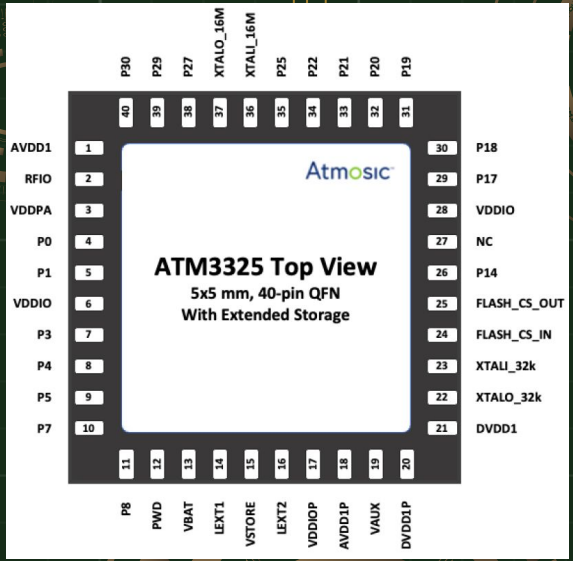
Model: NXP SR040 UWB Chip

Bluetooth Low Energy (BLE) Chip

Primary communication interface, ensuring compatibility and energy-efficient operation.

Key Features:

- **Power Efficiency:** Extremely low power consumption (0.85 mA in receive mode, 2.5 mA in transmit mode at 0 dBm).
- **Advanced Capabilities:** Supports Angle of Arrival (AoA) and Angle of Departure (AoD) for direction finding.
- **Security:** Features AES-256 encryption and hardware cryptographic accelerators.



Model: Atmosic ATM33 Series BLE SoC, compliant with Bluetooth 5.3

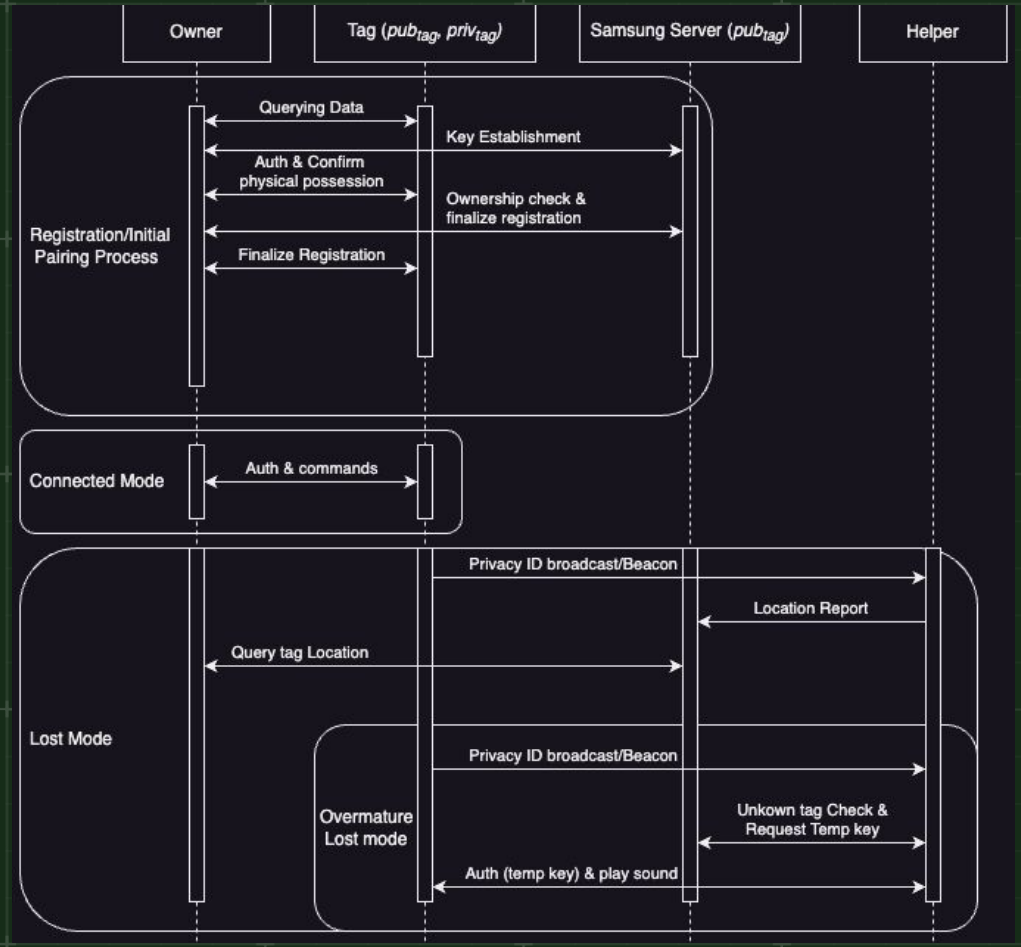


Figure: An overview of Samsung Offline Finding protocol for Smart-Tags

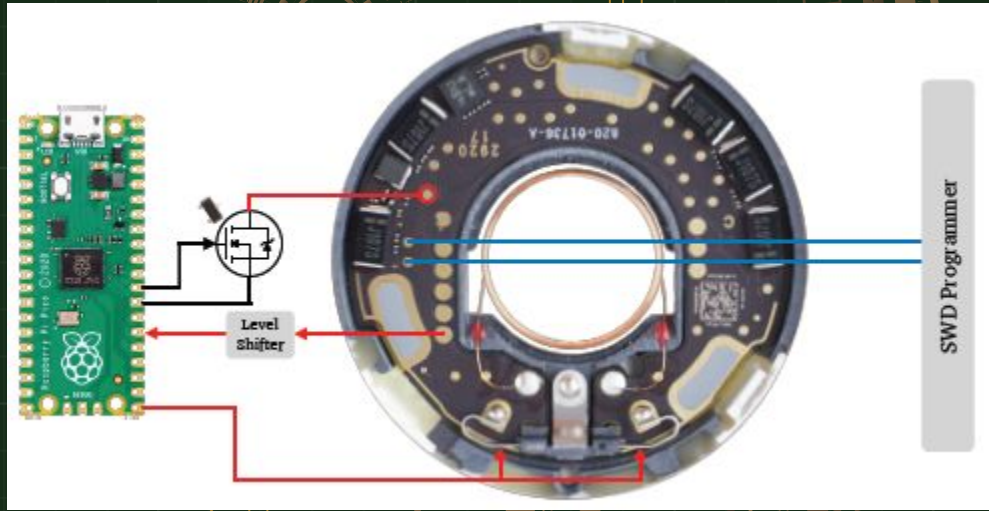
Credit: Tingfeng Yu and James Henderson and Alwen Tiu and Thomas Haines, "Privacy Analysis of Samsung's Crowd-Sourced Bluetooth Location Tracking System," ArXiv, vol. abs/2210.14702, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:253116608

3.

Related Work

Apple AirTag

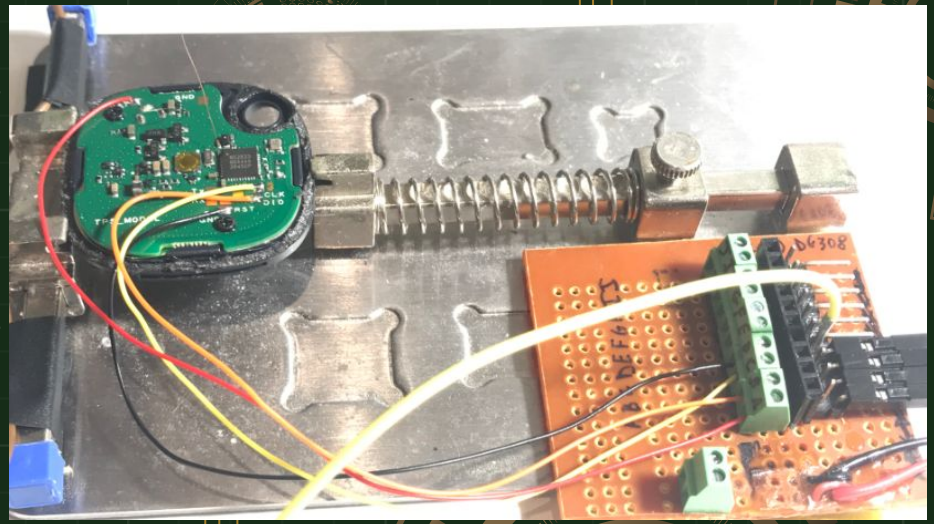
- Uncovered both hardware and software vulnerabilities in AirTags
- Firmware extraction via voltage glitching.
- Valuable insights into BLE vulnerabilities, helping us understand the security components of BLE devices before analyzing SmartTag 2.
- Ths provided insights into hardware-level security, informing our own research on SmartTag 2 by setting up a similar cost-effective probing environment.



Study: "AirTag of the Clones: Shenanigans with Liberated Item Finders"

Samsung SmartTag 1

- Undergone hardware-level reverse engineering, revealing vulnerabilities that allowed firmware dumping
- Provided detailed PCB images and descriptions of fault injection techniques used to bypass protections on the Nordic nRF52833 chip
- Identified test points for SPI and UART communication, showing the device's vulnerability to physical tampering



Github: "Samsung-SmartTag-Hack"

4.

Threat Model

Attacker Capabilities

- **Physical Access Attacker**
  - Can temper with PCB, modify or remove components
  - Extract firmware
- **Over-the-Air Attacker**
  - An attacker within range can exploit BLE or UWB signals.
  - Attacks could focus on communication vulnerabilities within the SmartTag 2's protocols.

Potential Threats

- **Stalking**
  - Attackers could covertly attach the SmartTag to a victim's belongings to track them.
- **Stolen Hardware**
  - Attackers could steal the SmartTag or the tagged item.
- **Firmware Readout and Modification**
  - Attackers with physical access could dump or modify firmware, bypassing encryption.
- **Identity Modification**
  - Altering a SmartTag's unique identifier could bypass security or conceal malicious use.

5.

Hardware Analysis

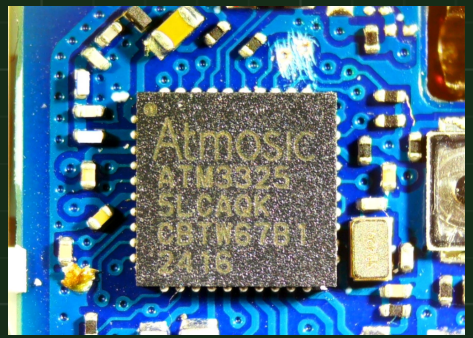
Hardware Analysis of Samsung SmartTag 2

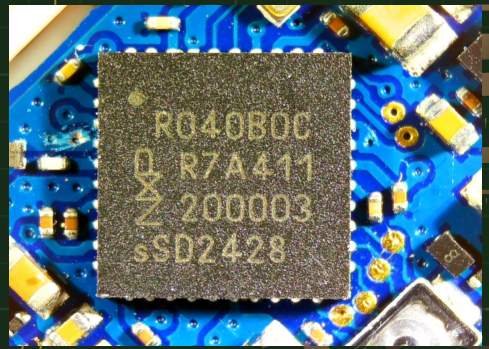
Objective:

- Investigate the hardware components of the Samsung SmartTag 2 to identify potential vulnerabilities through physical tampering and probing.

Chips Identified on PCB:

- NXP SR040: Ultra-Wideband (UWB) transceiver for precision ranging and secure data exchange.
- Atmosic ATM3325: BLE system-on-chip (SoC) for efficient communication and encryption.





Hardware Analysis of Samsung SmartTag 2

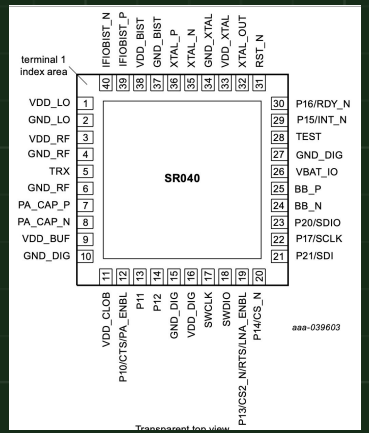
Investigating Open Test Pads

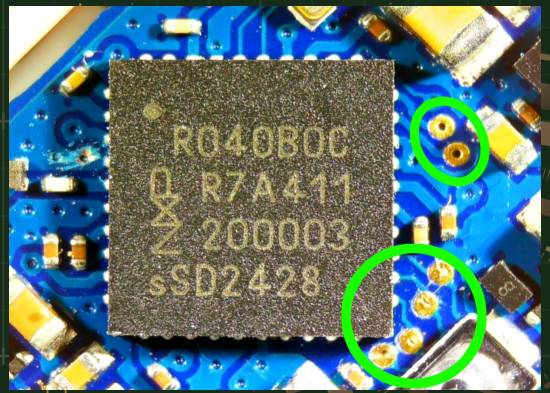
Observation:

- Four exposed SPI test pads on the NXP SR040 were identified (SCLK, SDI, SDO, CS).

Hypothesis:

- These pads could be used for interfacing, debugging, or programming.



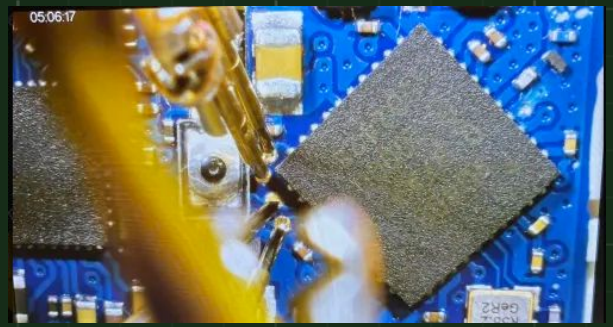


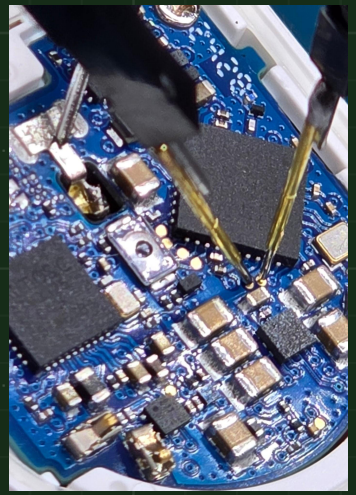
Hardware Analysis of Samsung SmartTag 2

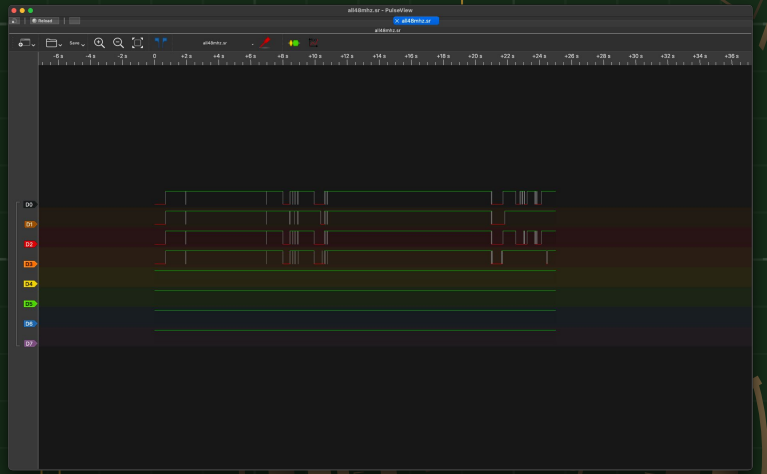
Investigating Open Test Pads

Findings:

- No significant output was detected during probing with a BitMagic Logic Analyzer.
- The SPI pins may be disabled or using a non-standard protocol.







Hardware Analysis of Samsung SmartTag 2

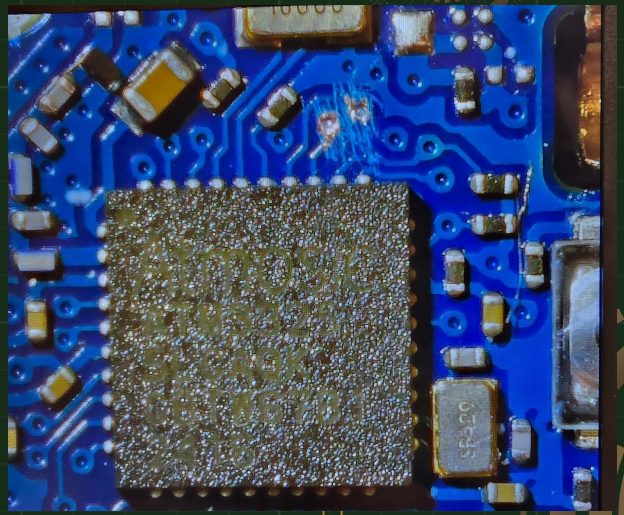
Probing Scratched Pads for UART

Experiment:

- Scraped PCB solder mask to expose potential UART pins on the Atmosic ATM33 chip.

Hypothesis:

- These pads could contain data as to firmware loading or communication between the BLE chip and the UWB chip.

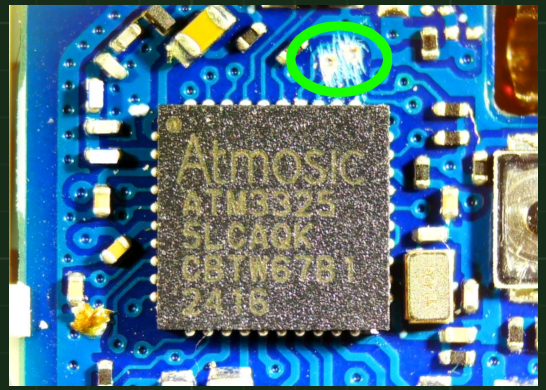


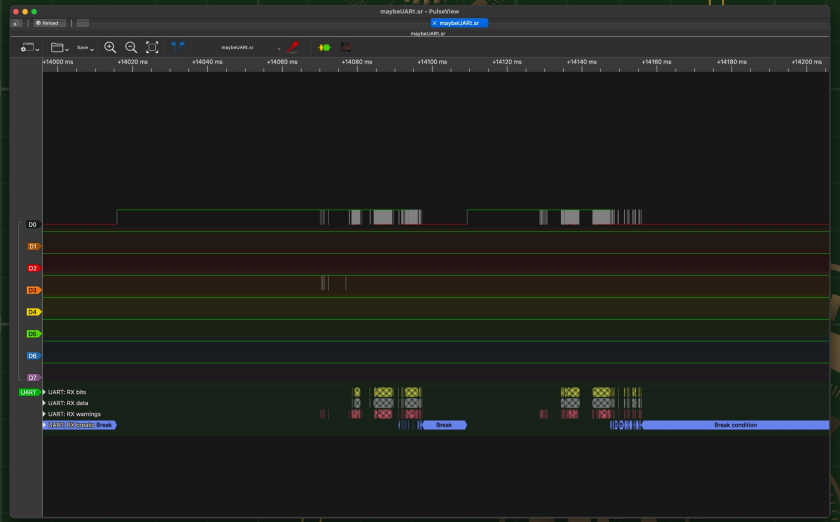
Hardware Analysis of Samsung SmartTag 2

Probing Scratched Pads for UART

Findings:

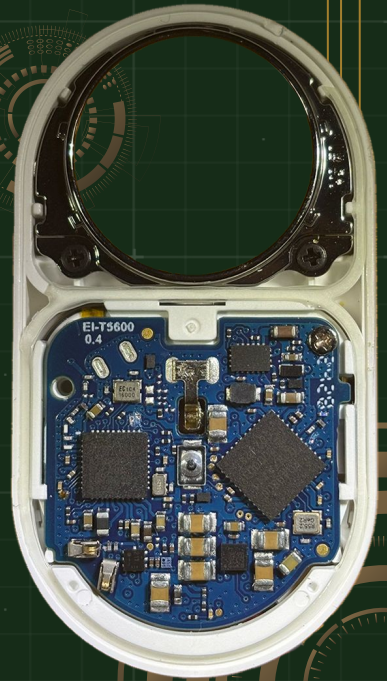
- Probed pins generated unrecognizable data, likely due to encryption.





Hardware Analysis Conclusion

We did not find any glaring security holes in the hardware implementation of the smarttag 2. We did notice that the device does function when powered on without the speaker attached, so could have stalking attack possibilities. Our expertise is limited so we are sure an more advanced researcher could find something exploitable from those two chips.



6.

Bluetooth Analysis

Bluetooth Analysis of Samsung SmartTag 2

Objective:

- To evaluate potential Bluetooth vulnerabilities in the Samsung SmartTag 2.
- Analyze pairing, command execution, and communication for exploitable weaknesses.

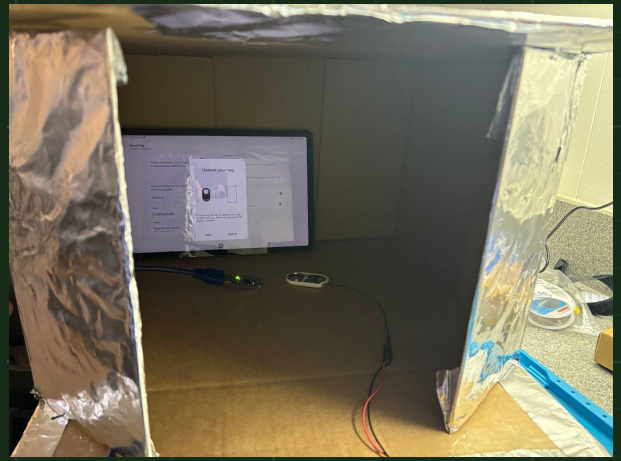
Tools Used:

- **NRF Dongle** connected to a **MacBook Pro** for capturing Bluetooth data.
- **Wireshark** for real-time packet analysis.
- **Android Debug Bridge (ADB)** for logging application-level interactions from a Samsung S8 tablet.

Initial Pairing Process:

- Captured **HCI-level Bluetooth commands** during pairing.
- Observed encrypted payloads and proprietary authentication protocols, which prevent unauthorized pairing attempts.
- No vulnerabilities found in the pairing process; the Bluetooth security stack appears secure.

Bluetooth Analysis of Samsung SmartTag 2



```
0000  01 52 0c f1 01 0f 09 61  6e 64 72 65 77 73 2d 74   ·R·····a ndrews-t
0010  61 62 6c 65 74 0b 03 0a  11 0c 11 0e 11 12 11 00   ablet··· ········
0020  12 01 05 01 07 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0030  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0040  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0050  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0060  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0070  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0080  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0090  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
00a0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
00b0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
00c0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
00d0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
00e0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
00f0  00 00 00 00 00                                     ·····
```

```
.../data/log/bt
monty@  [monty] $ cd Downloads/platform-tools
monty@  [platform-tools] $ ls
NOTICE.txt          hprof-conv          make_f2fs_casefold sqlite3
adb                 lib64               mke2fs
etc1tool            logs                mke2fs.conf
fastboot            make_f2fs           source.properties
monty@  [platform-tools] $ cd logs/log-2/FS/data/log/bt
monty@  [bt] $ ls
btcomms.log     btsnoop_hci.log
monty@  [bt] $
```

```
.../data/log/bt
```

```
btsnoop_hci.log
btatt
```

No.	Time	Source	Destination	Protocol	Length	Info
6389	178.1…	4c:87:46:56:00:20 ()	SamsungElect_2b:31:20 (andrews-tablet)	ATT	26	f05dc98937f88af1f4fb… Rcvd Read Response, Handle: 0x006b (Unknown)
6378	177.4…	SamsungElect_2b:31:20 (andrews-tablet)	4c:87:46:56:00:20 ()	ATT	12	Sent Read Request, Handle: 0x006b (Unknown)
6376	177.4…	4c:87:46:56:00:20 ()	SamsungElect_2b:31:20 (andrews-tablet)	ATT	10	Rcvd Write Response, Handle: 0x006c (Unknown)
6365	176.7…	SamsungElect_2b:31:20 (andrews-tablet)	4c:87:46:56:00:20 ()	ATT	14	0100 Sent Write Request, Handle: 0x006c (Unknown)
6359	176.7…	4c:87:46:56:00:20 ()	SamsungElect_2b:31:20 (andrews-tablet)	ATT	10	Rcvd Write Response, Handle: 0x0072 (Unknown)
6348	176.0…	SamsungElect_2b:31:20 (andrews-tablet)	4c:87:46:56:00:20 ()	ATT	14	0200 Sent Write Request, Handle: 0x0072 (Unknown)
6346	175.9…	4c:87:46:56:00:20 ()	SamsungElect_2b:31:20 (andrews-tablet)	ATT	10	Rcvd Write Response, Handle: 0x0092 (Unknown)
6330	175.3…	SamsungElect_2b:31:20 (andrews-tablet)	4c:87:46:56:00:20 ()	ATT	14	0200 Sent Write Request, Handle: 0x0092 (Unknown)
6328	175.2…	4c:87:46:56:00:20 ()	SamsungElect_2b:31:20 (andrews-tablet)	ATT	10	Rcvd Write Response, Handle: 0x0097 (Unknown)
6315	174.5…	SamsungElect_2b:31:20 (andrews-tablet)	4c:87:46:56:00:20 ()	ATT	14	0200 Sent Write Request, Handle: 0x0097 (Unknown)
6309	174.5…	4c:87:46:56:00:20 ()	SamsungElect_2b:31:20 (andrews-tablet)	ATT	10	Rcvd Write Response, Handle: 0x006c (Unknown)
6295	173.8…	SamsungElect_2b:31:20 (andrews-tablet)	4c:87:46:56:00:20 ()	ATT	14	0100 Sent Write Request, Handle: 0x006c (Unknown)
6294	173.8…	4c:87:46:56:00:20 ()	SamsungElect_2b:31:20 (andrews-tablet)	ATT	10	Rcvd Write Response, Handle: 0x0069 (Unknown)
6199	173.3…	SamsungElect_2b:31:20 (andrews-tablet)	4c:87:46:56:00:20 ()	ATT	14	0200 Sent Write Request, Handle: 0x0069 (Unknown)
5808	163.7…	SamsungElect_2b:31:20 (andrews-tablet)	4c:87:46:56:00:20 ()	ATT	10	Sent Handle Value Confirmation, Handle: 0x0068 (Unknown)
5807	163.7…	4c:87:46:56:00:20 ()	SamsungElect_2b:31:20 (andrews-tablet)	ATT	28	a0ee42c12f001e381964… Rcvd Handle Value Indication, Handle: 0x0068 (Unknown)
5806	163.7…	4c:87:46:56:00:20 ()	SamsungElect_2b:31:20 (andrews-tablet)	ATT	10	Rcvd Write Response, Handle: 0x0068 (Unknown)
5777	162.5…	SamsungElect_2b:31:20 (andrews-tablet)	4c:87:46:56:00:20 ()	ATT	28	080d432b43d28a72d11f… Sent Write Request, Handle: 0x0068 (Unknown)
5692	159.0…	SamsungElect_2b:31:20 (andrews-tablet)	4c:87:46:56:00:20 ()	ATT	10	Sent Handle Value Confirmation, Handle: 0x0068 (Unknown)
5691	159.0…	4c:87:46:56:00:20 ()	SamsungElect_2b:31:20 (andrews-tablet)	ATT	28	73e0066a8e423beb4dcb… Rcvd Handle Value Indication, Handle: 0x0068 (Unknown)
5689	159.0…	4c:87:46:56:00:20 ()	SamsungElect_2b:31:20 (andrews-tablet)	ATT	10	Rcvd Write Response, Handle: 0x0068 (Unknown)
5565	157.5…	SamsungElect_2b:31:20 (andrews-tablet)	4c:87:46:56:00:20 ()	ATT	28	800f1143f853cf77be1e… Sent Write Request, Handle: 0x0068 (Unknown)
5579	154.3…	4c:87:46:56:00:20 ()	SamsungElect_2b:31:20 (andrews-tablet)	ATT	10	Rcvd Write Response, Handle: 0x006c (Unknown)
5536	152.3…	SamsungElect_2b:31:20 (andrews-tablet)	4c:87:46:56:00:20 ()	ATT	14	0100 Sent Write Request, Handle: 0x006c (Unknown)
5377	146.4…	4c:87:46:56:00:20 ()	SamsungElect_2b:31:20 (andrews-tablet)	ATT	10	Rcvd Write Response, Handle: 0x009f (Unknown)
5293	145.2…	SamsungElect_2b:31:20 (andrews-tablet)	4c:87:46:56:00:20 ()	ATT	14	0200 Sent Write Request, Handle: 0x009f (Unknown)
4850	135.3…	SamsungElect_2b:31:20 (andrews-tablet)	4c:87:46:56:00:20 ()	ATT	10	Sent Handle Value Confirmation, Handle: 0x006e (Unknown)
4849	135.3…	4c:87:46:56:00:20 ()	SamsungElect_2b:31:20 (andrews-tablet)	ATT	28	565d98ed87b1614bc5d4… Rcvd Handle Value Indication, Handle: 0x006e (Unknown)
4776	130.9…	SamsungElect_2b:31:20 (andrews-tablet)	4c:87:46:56:00:20 ()	ATT	10	Sent Handle Value Confirmation, Handle: 0x0091 (Unknown)

```
> Frame 6389: 26 bytes on wire (208 bits), 26 bytes captured (208 bits)
> Bluetooth
> Bluetooth HCI H4
> Bluetooth HCI ACL Packet
> Bluetooth L2CAP Protocol
> Bluetooth Attribute Protocol
```

```
0000  02 05 20 15 00 11 00 04  00 0b f0 5d c9 89 37 f8   ·· ···· ···]··7·
0010  8a f1 f4 fb 24 93 ca ed  1d f5                     ····$··· ··
```

Bluetooth Analysis of Samsung SmartTag 2

Reset Command Hypothesis:

- A Reset command was detected during pairing:
  - Potential attack vector: If the reset command is repeatedly sent, it could render the device non-functional.
  - Attempts to exploit this were unsuccessful due to encryption mechanisms and inability to verify command execution.

```
Frame 1: 4 bytes on wire (32 bits), 4 bytes captured (32 bits)
Bluetooth
    [Source: host]
    [Destination: controller]
Bluetooth HCI H4
    [Direction: Sent (0x00)]
    HCI Packet Type: HCI Command (0x01)
Bluetooth HCI Command - Reset
```

Security Challenges:

- **Encryption and proprietary protocols** make unauthorized interactions difficult.
- Dependency on the **Samsung application** adds an extra layer of protection, preventing bypass with standard tools.

Bluetooth Analysis of Samsung SmartTag 2

Future Research Directions:

- Explore advanced BLE transmission tools to bypass authentication mechanisms.
- Perform hardware-level debugging (e.g., using JTAG or UART) to gather more insights into the SmartTag 2's response to reset commands.
- Develop a custom BLE stack to mimic the Samsung tablet and test command interactions directly.

Conclusion:

- No definitive exploits were found, but the analysis shows complexity in BLE implementations.
- Continued research is needed with more advanced tools and techniques to uncover potential vulnerabilities.

7.

Conclusion

Key Findings of Our Analysis

Hardware:

- Test pads and scratched PCB pins did not yield usable data due to encryption or disabled interfaces.
- The speaker poses a potential privacy risk, allowing for silent tracking. Since the tracker powers on and works with it disconnected.

Bluetooth:

- Proprietary encryption and authentication mechanisms prevented successful payload manipulation.
- The Reset command hypothesis showed potential for disruption but could not be verified without bypassing encryption.

Conclusion Overview

Research Summary:

- Our investigation into the Samsung SmartTag 2 highlighted the robustness of its security features, both in hardware and Bluetooth communication.

Main Focus Areas:

- Explored hardware vulnerabilities through PCB analysis and test pad probing.
- Evaluated Bluetooth communication for potential exploits, such as command injection or payload manipulation.

Takeaway:

- BlackBox hardware testing is not as easy as in the movies.
- While no definitive exploits were uncovered, the research sets a foundation for further exploration of IoT device vulnerabilities.

Thank You!