



CVE-2026-31280: Insecure Bluetooth "RFCOMM" Leading to Device Crash in Parani M10 Intercom



Device: Parani M10 Intercom Firmware Version:V2.1.3 Company Name: Parani India (Backed by Sena technologies Inc.)

1 Security Assessment Details

1.1 Executive Summary

This vulnerability impacts device availability, integrity and confidentiality due to the exposure of an unauthenticated Bluetooth Classic RFCOMM service combined with improper input validation. The device allows unauthenticated connection by bypassing of enabling pairing mode within Bluetooth range to establish a session and send unauthorized commands. A threat actor can disrupt normal operation by injecting arbitrary audio or sending malicious input that triggers Man-In-Middle(MITM), buffer overflow which are leads to device crash, disrupt authorized Bluetooth connection.

1.2 Scope and Objectives

The scope of this assessment was limited to Bluetooth Classic Communication of Parani M10 Intercom(Firmware Version 2.1.3).

1.3 Technology Impact Summary

The security assessment of the Bluetooth Classic communication interface has been performed. These assessments aimed to identify security weaknesses in the evaluated intercom device, explain the impact and associated risks, and provide guidance for prioritization and remediation.

Following are the technical impacts:

An attacker within Bluetooth range can connect to an exposed RFCOMM service without authentication and perform unauthorized actions, including sending of arbitrary payloads leads to device crash until next manual intervention. Also attacker do MITM causes to play arbitrary audio discontinue authorized connection between user mobile Intercom device.

1.4 Business Impact Summary

1.4.1 Overall Business Impact

The identified wireless-layer vulnerability represents a significant high business risk, as successful exploitation could disrupt normal device operation, allow unauthorized malicious data injection, negatively impact customer trust and user experience, damage brand reputation and market credibility, introduce safety concerns in sensitive deployment environments, and potentially expose the organization to regulatory scrutiny, legal liability, and financial costs associated with remediation and customer support.

1.5 Testing Environment and Tools

Bluetooth security testing was conducted using standard Bluetooth enumeration and interaction tools:

- The intercom device equipped with Bluetooth Classic functionality
- A Linux-based security testing workstation (Kali Linux) with TP -Link Bluetooth adapter for Bluetooth connectivity testing.
- "bluetoothctl" CLI tool for pairing analysis and service enumeration.
- "RFCOMM" utilities used for establish direct channel communication and transmit crafted test payloads.
- "btmon" utility used to capture logs for analysis and service enumeration.
- A python Script for payloads flooding.

1.6 Table of Findings

Finding ID	Scope	Finding	CVSS Score	CVSS Vector	Severity	Status
PM10-IBT-01	Bluetooth Classic	Insecure Bluetooth Communication	8.6	CVSS:4.0 / AV:A / AC:L / AT:N / PR:N / UI:P / VC:H / VI:H / VA:H / SC:N / SI:N / SA:N	High	Not Fixed

1.7 Device Strengths

Not assessed in this engagement (scope limited to Bluetooth Classic communication). No additional strengths were evaluated.

1.8 Device Weaknesses

- The intercom device exposes an unauthenticated Bluetooth Classic RFCOMM service, allowing unauthorized nearby attackers to establish a connection bypassing of enabling pairing mode on the device. This leads to create buffer overflow conditions further that can result in device crash or reboot until manual intervention.

- The exposure of RFCOMM channels (CH10, CH12) in the device makes attacker to create a MITM environment. It discontinues authorized user establishment which disrupts Bluetooth communications by playing arbitrary noise or crashing the device.

2 Technical Findings

2.1 PM10-IBT-01: Insecure Bluetooth Communication

Potential Impact: High

Description

The intercom device relies on Bluetooth Classic communication to provide local wireless functionality. During the assessment, it was observed that the device exposes an RFCOMM service without requiring pairing or authentication. A threat actor within Bluetooth range can establish a direct connection to the device and transmit unauthorized commands. It was further identified that the RFCOMM service does not properly validate input length. By sending malicious data, the attacker can trigger a buffer overflow condition, causing the device to crash or reboot. Additionally, unauthorized audio playback can be performed, disrupting normal device behavior.

Affected Components

- The device's Bluetooth Classic communication module.
- The exposed RFCOMM service interface.
- The command-processing and input-handling mechanism within the firmware.

Technical Risk

An attacker can establish an unauthenticated connection and inject arbitrary commands, impacting device integrity. Additionally, exploitation of the buffer overflow condition can cause the device to become unavailable resulting in Denial-of-Service state until physical intervention to restore normal functionality.

Business Risk

Exploitation of this vulnerability could:

- Lead to operational downtime or instability of the device
- Increase customer dissatisfaction and operational support costs
- Negatively affect brand reputation and long-term consumer trust

Steps to Reproduce (High-Level) Case 1 - Buffer Overflow Attack

1. Perform OSINT on Target Device: Conduct open-source intelligence (OSINT) to gather publicly available information related to the intercom device, including identification of its Bluetooth MAC address.
2. Validate Device MAC Address: Physically inspect the intercom device and confirm that the Bluetooth MAC address identified during OSINT matches the MAC address printed on the back label of the device.



3. Assess Pairing Configuration: Attempt to pair with the device and observe that it allows pairing without requiring authentication or user verification.
4. Enumerate Available Services Post-Pairing: After establishing the unauthenticated pairing, review the list of exposed Bluetooth services and identify that an RFCOMM service is active on channel 10 and 12.

```

Count: 1
#39: len 26 (9 Kb/s)
Latency: 21 msec (6-27 msec -22 msec)
Channel: 20 [PSM 1 mode Basic (0x00)] [chan 0]
Channel Latency: 21 msec (21-24 msec -22 msec)
ACL Data RX: Handle 3 flags 0x02 dlen 31 #41 9.853438
Channel: 64 len 27 [PSM 1 mode Basic (0x00)] [chan 0]
SDP: Service Search Attribute Response (0x07) tid 2 len 22
Attribute bytes: 19
Continuation state: 0
Combined attribute bytes: 95
Attribute list: [len 89] [position 0]
Attribute: Service Record Handle (0x0000) [len 2]
0x0010000
Attribute: Service Class ID List (0x0001) [len 2]
UUID (3) with 2 bytes [0 extra bits] len 3
Handsfree (0x111e)
UUID (3) with 2 bytes [0 extra bits] len 3
Generic Audio (0x1203)
Attribute: Protocol Descriptor List (0x0004) [len 2]
Sequence (6) with 3 bytes [8 extra bits] len 5
UUID (3) with 2 bytes [0 extra bits] len 3
L2CAP (0x0100)
Sequence (6) with 3 bytes [8 extra bits] len 7
UUID (3) with 2 bytes [0 extra bits] len 3
RFCOMM (0x0003)
Unsigned Integer (1) with 1 byte [0 extra bits] len 2
0x00
Attribute: Language Base Attribute ID List (0x0006) [len 2]
Unsigned Integer (1) with 2 bytes [0 extra bits] len 3
0x056e
Unsigned Integer (1) with 2 bytes [0 extra bits] len 3
0x006a
Unsigned Integer (1) with 2 bytes [0 extra bits] len 3
0x0100
Attribute: Bluetooth Profile Descriptor List (0x0009) [len 2]
Sequence (6) with 6 bytes [8 extra bits] len 8
UUID (3) with 2 bytes [0 extra bits] len 3
Handsfree (0x111e)
Unsigned Integer (1) with 2 bytes [0 extra bits] len 3
0x0107
Attribute: Unknown (0x0100) [len 2]
Hands-Free unit [len 15]
Attribute: Unknown (0x0311) [len 2]
0x00df
HCI Command: Authentication Requested (0x01|0x0011) plen 2 #42 9.853705
Handle: 3 Address: 00:01:95:78:CF:25 (Sena Technologies, Inc.)
HCI Event: Command Status (0x0f) plen 4 #43 9.857006
Authentication Requested (0x01|0x0011) ncmd 2
Status: Success (0x00)

```

```

kali@kali:~$ bluetoothctl
[bluetoothctl]
[New] Media7Org/Bluez/hci0
SupportedUUIDs: 000011a-0000-1000-0000-00005f9b34fb
SupportedUUIDs: 000011b-0000-1000-0000-00005f9b34fb
Agent registered
[bluetoothctl] connect 00:01:95:78:CF:25
Attempting to connect to 00:01:95:78:CF:25
[CHG] Device 00:01:95:78:CF:25 Connected: yes
Connection successful
[CHG] Device 00:01:95:78:CF:25 ServicesResolved: yes
[Parani M10]>

```

5. Establish RFCOMM Connection: Initiate a direct RFCOMM connection to channel 10 using appropriate Bluetooth utilities.

```

= Index Info: DC:62:79:C4:49:44 (Realtek Semiconductor Corporation)
bluetoothd[107977]: @ MGMT Open: bluetoothd (privileged) version 1.23
< HCI Command: Create Connection (0x01|0x0005) plen 13
Address: 00:01:95:78:CF:25 (Sena Technologies, Inc.)
Packet type: 0xcc18
DM1 may be used
DH1 may be used
DM3 may be used
DH3 may be used
DM5 may be used
DH5 may be used
Page scan repetition mode: R2 (0x02)
Page scan mode: Mandatory (0x00)
Clock offset: 0x0000
Role switch: Allow peripheral (0x01)
> HCI Event: Command Status (0x0f) plen 4
Create Connection (0x01|0x0005) ncmd 2
Status: Success (0x00)
> HCI Event: Connect Complete (0x03) plen 11
Status: Success (0x00)
Handle: 1
Address: 00:01:95:78:CF:25 (Sena Technologies, Inc.)
Link type: ACL (0x01)
Encryption: Disabled (0x00)
< HCI Command: Read Remote Supported Features (0x01|0x001b) plen 2
Handle: 1 Address: 00:01:95:78:CF:25 (Sena Technologies, Inc.)
> HCI Event: Command Status (0x0f) plen 4
Read Remote Supported Features (0x01|0x001b) ncmd 2
Status: Success (0x00)
> HCI Event: Max Slots Change (0x1b) plen 3
Handle: 1 Address: 00:01:95:78:CF:25 (Sena Technologies, Inc.)
Max slots: 5
> HCI Event: Max Slots Change (0x1b) plen 3
Handle: 1 Address: 00:01:95:78:CF:25 (Sena Technologies, Inc.)
Max slots: 5
> HCI Event: Read Remote Supported Features (0x0b) plen 11
Status: Success (0x00)
Handle: 1 Address: 00:01:95:78:CF:25 (Sena Technologies, Inc.)
Features: 0xff 0xfe 0x8f 0xfe 0xd8 0xff 0x5b 0x83
3 slot packets
5 slot packets
Encryption
Slot offset
Timing accuracy
Role switch
Hold mode
Sniff mode
Power control requests
Channel quality driven data rate (CQDDR)
SCO link
HV2 packets

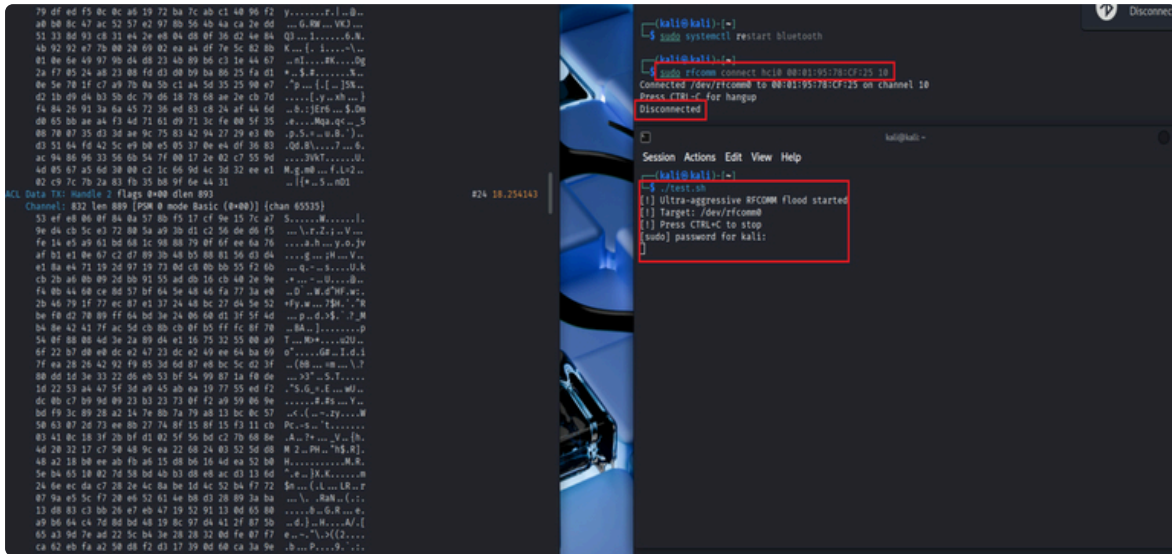
```

```

kali@kali:~$ sudo rfcomm connect hci0 00:01:95:78:CF:25 10
[sudo] password for kali:
Connected /dev/rfcomm0 to 00:01:95:78:CF:25 on channel 10
Press CTRL-C for hangup

```

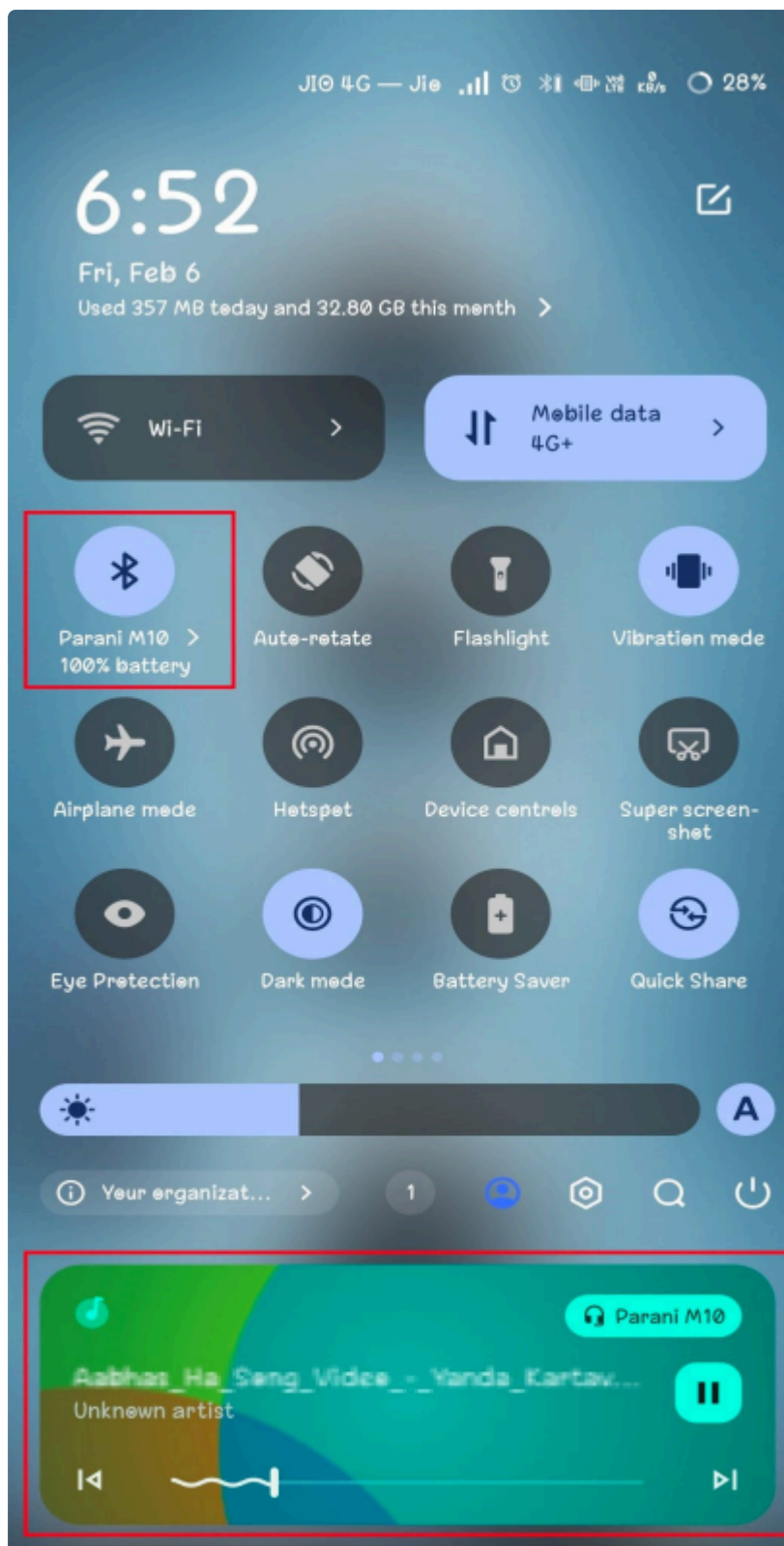
- 6. Execute Crafted Script: Run a specially crafted script to transmit unauthorized and oversized input data through the established RFCOMM session.



- 7. Observe that the device accepts the injected input and subsequently crashes or powered off, confirming the presence of an unauthenticated command injection vulnerability combined with a buffer overflow condition.

Case 2- Man In the Middle Attack

- 1. Connect the mobile phone to the intercom via Bluetooth and begin playing an audio track to confirm a stable connection.



2. Audio data was injected into the intercom stream through the Paplay utility. The attack succeeded, resulting in arbitrary audio being played over the intercom through an unauthorized method.

- 5. It was observed that the mobile device was successfully disconnected from the intercom, and the intercom became unavailable for further connections until manual intervention is performed.

```
FCS: 0x0e
3a fd c8 f2 68 f6 1a 87 f9 10 f0 e0 10 06 48 21 1...N.....M!
56 19 27 97 09 c0 11 91 ff 09 24 17 4f f9 a5 6c V.....*O..l
2a 06 0a 10 50 30 7c af 9c 90 4a 74 d2 ba f4 ea *T:8!...J.....
72 a1 cd 6a 7e 28 5b c8 21 42 ba 8b ee b8 8f 8a 7...J-(P..B.....
be 00 a1 c5 ee 0c 18 c3 27 0f 00 40 69 29 91 .J.....M!P.
bd c1 f1 40 97 76 46 81 01 58 f5 01 43 52 0d 0b .....*P...C.
vd 32 44 b0 07 05 ad ea 7a 4a 2a 78 c3 30 2e 20 .J.....2N.p.i.*
82 90 0a ee 05 7a 10 5a 02 43 03 30 ea c8 c9 00 --T..*..*.....

#185 48.804668
#186 48.873806
#187 48.874170
#188 48.882633

Num handles: 1
Handles: 2 Address: 00:01:95:78:CF:25 (Sena Technologies, Inc.)
Count: 1
#187: len 135 [100 Kb/s]
Latency: 6 msec (4-10 msec -0 msec)
Channel: 192 [PSM 3 mode Basic (0x00)] [chan 1]
Channel latency: 0 msec (4-10 msec -0 msec)
ACL Data RX: Handle 2 flags 0x02 dlen 9 #186 48.873806
Channel: 6A len 5 [PSM 3 mode Basic (0x00)] [chan 1]
RFCOMM: Unconnected info with header check (UID) (0xef)
Address: 0x01 cr 0 dlc1 0x18
Control: 0xef poll/final 1
Length: 4
FCS: 0xc8
Credits: 1
ACL Data TX: Handle 2 flags 0xb0 dlen 135 #187 48.874170
Channel: 192 len 111 [PSM 3 mode Basic (0x00)] [chan 1]
RFCOMM: Unconnected info with header check (UID) (0xef)
Address: 0x03 cr 1 dlc1 0x10
Control: 0xef poll/final 0
Length: 127
FCS: 0x0e
0a 62 ka 09 92 78 f6 39 0a 2f 21 09 7d 9a 7f 3b .b2].k.9./i...j
09 0e 77 13 12 40 00 0d 26 cf 09 09 44 07 09 cf --W!M..0...M...
02 77 15 06 09 01 0c 2f 3a 5a f5 00 42 44 f2 5a --m.d.a...I...02.2
90 3c 65 c5 f9 f3 99 96 5b 2c 4c 51 49 35 44 43 .4e....V.IQIS.C
48 1f 38 2c 05 00 09 30 20 07 00 11 23 14 c8 c7 B..*...-gI....
99 be 06 00 00 25 05 15 e6 d3 bd 5a 07 00 05 55 --V.0a....*...0
c1 08 d2 04 7d 01 dd be 3a 20 11 8a 3a c5 5d .....|a..60...|
13 ff 04 01 ad 02 4e c1 45 50 12 a3 3a 00 0e .....*..k.....

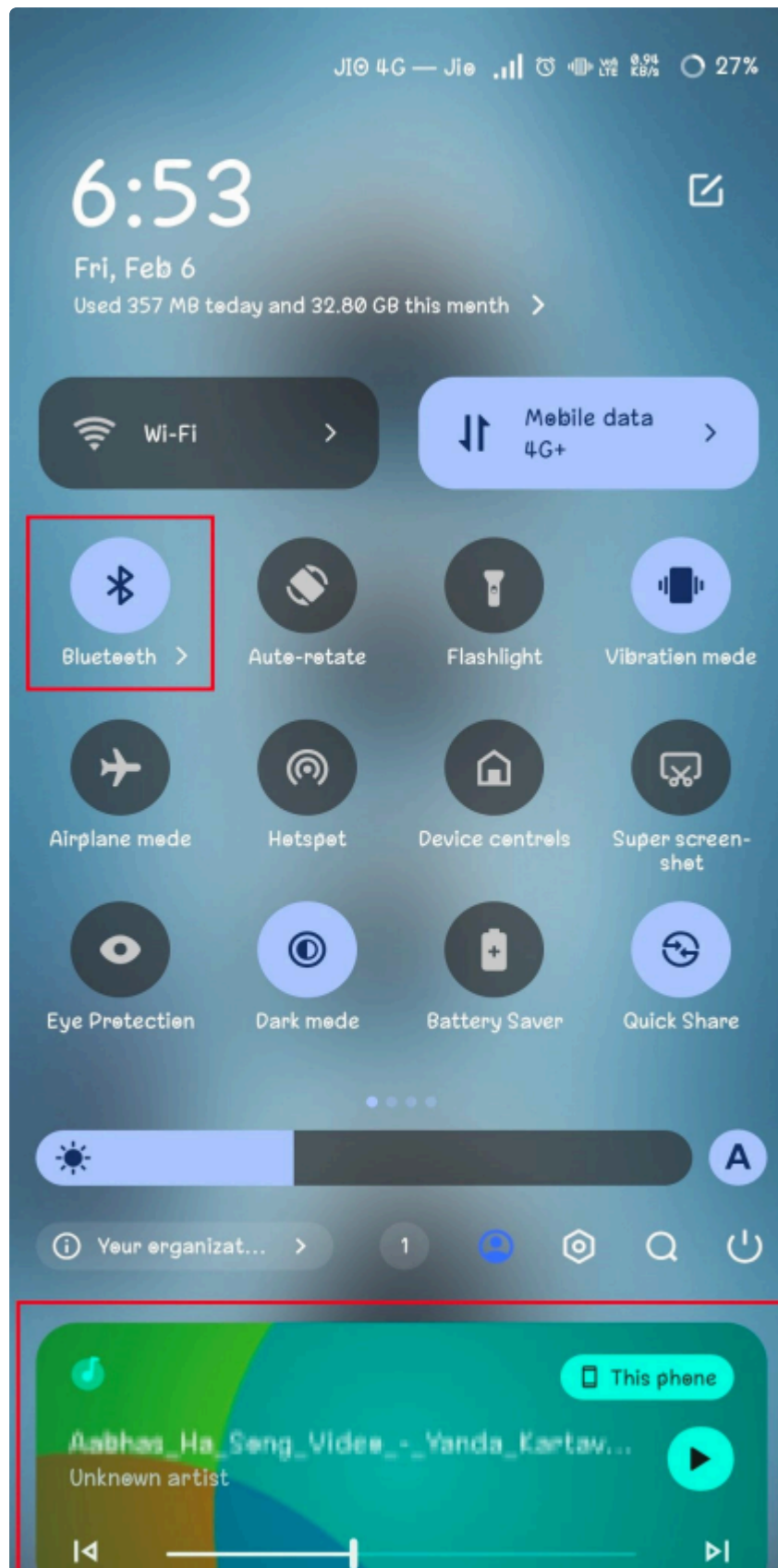
Num handles: 1
Handles: 2 Address: 00:01:95:78:CF:25 (Sena Technologies, Inc.)
Count: 1
#187: len 135 [100 Kb/s]
Latency: 6 msec (4-10 msec -0 msec)
Channel: 192 [PSM 3 mode Basic (0x00)] [chan 1]

[hal@kali: ~]$ sudo ./test.sh
[!] Ultra-aggressive RFCOMM flood started
[!] Target: /dev/rfcomm
[!] Press CTRL+C to stop

[hal@kali: ~]$ hciconfig
hci0: Type: Primary Bus: USB
BD Address: DC:62:79:C4:69:44 ACL MTU: 10216 SCO MTU: 1024
BTUUID: 00000000-0000-1000-8000-000000000000
UP RUNNING
RX bytes:29907 acl:548 sco:0 events:1444 errors:0
TX bytes:82142 acl:535 sco:0 commands:740 errors:0

[hal@kali: ~]$ sudo rfcomm connect hci0 00:01:95:78:CF:25 12
Connected /dev/rfcomm0 to 00:01:95:78:CF:25 on channel 12
Press CTRL-C for hangup
Disconnected

[hal@kali: ~]$
```



Notes

This proof-of-concept was conducted in a controlled and authorized testing environment.

The assessment was limited to demonstrating unauthenticated command injection and service disruption through a buffer overflow condition over Bluetooth Classic communication. No attempts were made to perform advanced exploitation beyond validating the identified vulnerability.

3 Remediation and Recommendations

- Enforce Secure Authentication and Pairing Controls: Configure the Bluetooth interface to require authenticated pairing (e.g., Secure Simple Pairing with user confirmation) before granting access to any services.
- Implement Service-Level Authorization: Restrict access to the RFCOMM service by validating authenticated link keys and enforcing authorization checks prior to processing commands.
- Strengthen Input Validation Mechanisms: Implement strict bounds checking and proper input length validation within the RFCOMM service handler to prevent buffer overflow conditions.

4 Appendix — Contact and Disclosure

Researcher(s): Tapadyuti Baral, Nikhil Yalgar

Contact: arghyabaral@gmail.com, nik.sec127001@proton.me

[Previous](#)
[Finding Hardcoded Telnet Credentials](#)

Last updated 29 days ago