Providing Seamless Secure Protocols for Airspace Future Communications Infrastructure

Jubran Altaweel, Saranya Krishna Kumar, Hannes Widéen, Max Wilén, Elis Öhman

I. Introduction

The **Host Identity Protocol (HIP)** is an advanced networking protocol designed to enhance security, mobility, and flexibility in IP-based networks. HIP achieves this by separating the traditional roles of IP addresses into two distinct functions: host identities and locators. In this framework, each host is assigned a cryptographic identity, known as a Host Identity (HI), which remains constant regardless of the host's network location. This HI is typically represented by a public key, providing a unique and verifiable identity for secure communication between devices. On the other hand, IP addresses are redefined as locators, which indicate the current network location of a host, guiding the routing of data packets. This separation allows HIP to offer significant advantages in dynamic and mobile environments, where hosts may frequently change their network locations. HIP's use of cryptographic identities ensures enhanced security, as hosts are authenticated through Host Identity Tags (HITs), and communication is often encrypted using IPsec.

Moreover, HIP supports both mobility and multi-homing, enabling hosts to move between networks or connect to multiple networks simultaneously without disrupting ongoing communication sessions. This makes HIP particularly valuable for applications requiring secure, continuous communication across diverse and changing network environments. Importantly, HIP is designed to be compatible with existing IP-based networks, allowing for gradual integration and deployment without requiring extensive changes to the current infrastructure. Overall, HIP provides a robust and flexible solution for secure and resilient networking in modern, dynamic environments.

Future Communication Infrastructure (FCI) is envisioned as the next generation of aeronautical communication systems, integrating diverse communication technologies such as **LDACS (L-band Digital Aeronautical Communication System)**, **Aeronautical Mobile Airport Communication System (AeroMACS)**, and **Satellite Communications (SATCOM)**. These systems must work together to provide continuous, secure communication for aircraft as they transition between different airspaces and network coverage areas. Managing this seamless transition and ensuring secure communication across diverse technologies presents significant challenges.

II. Project description

This project focuses on implementing and evaluating the **HIP** within the **Future Communication Infrastructure (FCI)** framework, specifically for use in aeronautical communication systems. The project will leverage HIP's capabilities to manage secure, seamless handovers and multi-homing across multiple communication technologies, including **LDACS**, **AeroMACS**, and **SATCOM**.

Two network simulators/emulators were considered to be used for the project, **NS-3** and **Common Open Research Emulator (CORE)**) The one that decided to be used in the project was **NS-3**. The reason behind that decision was that **NS-3** is a simulator and **CORE** is an emulator because the project's focus was on controlled, scalable, and reproducible simulations of network protocols, where the real-time hardware integration aspects of an emulator were not required. NS-3's strengths in protocol modeling, scalability, and customizability aligned better with the project's needs than CORE's real-time emulation capabilities. Furthermore, if the simulation works well in NS-3, CORE could be used with the same scenario.

A. Project goals

The goals of the projects can be divided into three categories.

1) Seamless Handover:

- Implement HIP to manage seamless handovers between different communication technologies within FCI
- Simulate scenarios where aircraft transition between networks like SATCOM, LDACS, and AeroMACS and evaluate HIP's effectiveness in maintaining continuous communication without interruptions.

2) Multi-Homing Support:

- Enable HIP's multi-homing capabilities, allowing aircraft to connect to multiple networks simultaneously and dynamically switch between them based on real-time conditions and Quality of Service (QoS) requirements.
- Test and evaluate HIP's ability to manage multiple network connections, ensuring optimal communication performance.

3) Secure Communication:

- Ensure all communications within the HIP-FCI environment are secure, using HIP's cryptographic methods to dynamically identify and authenticate hosts.
- Protect data integrity and maintain secure communication channels during network transitions and potential threat scenarios.

B. Limitations

This project focuses on implementing and evaluating the **HIP** within the **FCI** framework, specifically for use in aeronautical communication systems. The project will leverage HIP's capabilities to manage secure, seamless handovers and multi-homing across multiple communication technologies, including **LDACS**, **AeroMACS**, and **SATCOM**.

Two network simulators/emulators were considered for the project: **NS-3** and **Common Open Research Emulator (CORE)**. The chosen tool for the project was **NS-3**. The decision was based on the fact that **NS-3** is a simulator, whereas **CORE** is an emulator. Since the project's focus is on controlled, scalable, and reproducible simulations of network protocols, the real-time hardware integration aspects of an emulator were not required. NS-3's strengths in protocol modeling, scalability, and customizability aligned better with the project's needs compared to CORE's real-time emulation capabilities. Furthermore, if the simulation performs well in NS-3, CORE could be utilized for testing the same scenarios in a real-time environment.

C. Project Goals

The project goals are categorized into three main objectives:

1) Seamless Handover:

- **Implement HIP** to manage seamless handovers between different communication technologies within FCI.
- Simulate scenarios where aircraft transition between networks like SATCOM, LDACS, and AeroMACS and evaluate HIP's effectiveness in maintaining uninterrupted communication.

2) Multi-Homing Support:

- Enable HIP's multi-homing capabilities, allowing aircraft to connect to multiple networks simultaneously and dynamically switch between them based on real-time conditions and Quality of Service (QoS) requirements.
- Test and evaluate HIP's ability to manage multiple network connections, ensuring optimal communication performance.

3) Secure Communication:

- Ensure all communications within the HIP-FCI environment are secure, leveraging HIP's cryptographic methods to dynamically identify and authenticate hosts.
- Protect data integrity and maintain secure communication channels during network transitions and potential threat scenarios.

D. Limitations

The project starts on September 9, 2024, and concludes on December 16, 2024. It is part of the TDDE63 Advanced Project Course: Information Security, with a workload equivalent to 6 academic credit points for each project member. Each 1.5 academic credit points represents approximately 40 hours of work, giving each project member around 160 hours to complete the project.

III. Background

This section delves into the Future Communication Infrastructure (FCI), a cutting-edge communication system designed for next-generation aviation. It examines the airborne and ground-based elements that enable secure and seamless communication between aircraft and ground stations. Additionally, it provides an in-depth analysis of HIP, outlining its key components and its critical role in enhancing the security and mobility of modern communication systems in aviation.

A. Future Communication Infrastructure

The FCI is an advanced IP-based global communication system designed to enable seamless Air-Ground and Ground-Ground communications. It integrates broadband data links, such as SATCOM, LDACS, and AeroMACS, to provide enhanced connectivity and robust communication channels across the global aviation network.

1) Airborne Components: The airborne segment of the FCI consists of several critical components that ensure efficient and secure communication between aircraft and ground stations:

- **Airborne Router (A-R):** A critical Layer 3 device within the aircraft, responsible for link selection and managing inter-technology handovers. It ensures uninterrupted communication by switching between available networks.
- **Airborne Radios (AR):** Also known as Airborne Stations, these devices enable communication over the air. They comply with Air/Ground subsystem standards and interface with the Airborne Router at Layer 2 to facilitate data exchange with ground systems.
- **Airborne End System (A-E):** Positioned above Layer 3, this system manages transport and application services within the aircraft, ensuring efficient and secure communication services for onboard systems and applications.

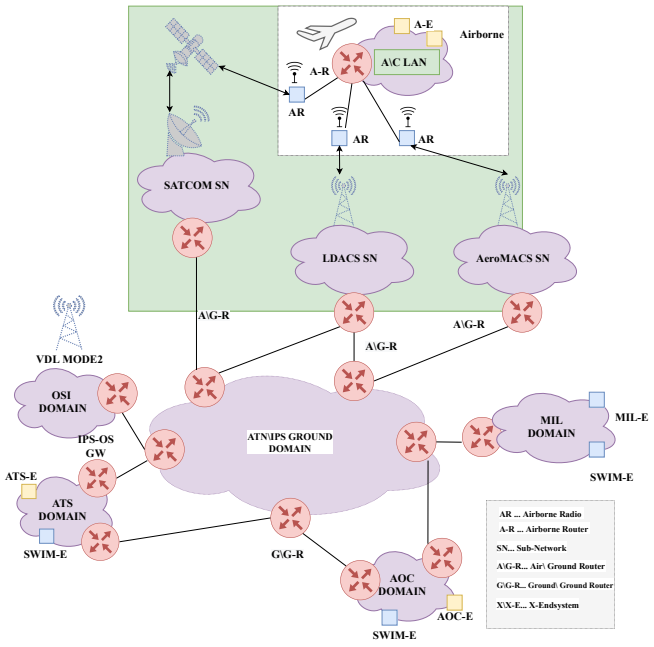

Fig. 1. The Future Communications Infrastructure.

2) Ground Components: The ground segment of the FCI features various subsystems that provide the necessary infrastructure for comprehensive communication capabilities. These components include:

1) **Access Subnets and Ground Radios:**
   - **Ground Radios (GR):** Essential for facilitating ground-to-air communications, these devices implement the Air/Ground subsystem standards and provide a Layer 2 interface with routers.
   - **Sub-Networks (SN):** Typically IP-based, these networks connect access providers and ground subsystems. A notable exception is VDL Mode 2, which uses ATN-OSI for internal communications.
   - **Air/Ground Router (A/G-R):** This router enables the seamless transfer of data between airborne and ground networks, ensuring a reliable and continuous exchange of information.
2) **Ground/Ground Router (G/G-R):** Serving as a boundary router, this component manages data transmission between ground networks, ensuring reliable Ground-Ground communication within the infrastructure.
3) **OSI-IPS Gateways:** These gateways bridge legacy B1 and B2 communication services with the new FCI architecture, using the OSI protocol to integrate existing systems with modern IP-based solutions.

3) Ground End Systems: The FCI also incorporates ground-end systems (X-E) that handle higher-layer transport and application functions crucial for air traffic management. These include:

- **End Systems for Air Traffic Services:** Located in Air Traffic Control (ATC) centers and airports, these systems provide essential communication services for civil and military operations.
- **AOC End Systems:** Designed specifically for Airline Operational Communications (AOC), these systems support exchanging critical operational data necessary for airline management and coordination.

This advanced network architecture significantly enhances the scalability, flexibility, security, and reliability of air traffic management communications, meeting the growing demands of modern aviation.

4) Very High-Frequency Data Link (VHF): VHF communication has been a cornerstone of aviation for decades, providing the primary voice communication link between pilots and ATCs. Operating within the frequency range of 118.00 to 136.975 MHz, VHF enables clear, real-time conversations crucial for time-sensitive and immediate instructions, such as takeoff and landing clearances. However, with the rise in global air traffic, particularly around busy airports, VHF channels are increasingly congested. To alleviate this, some regions have reduced the spacing between VHF channels to accommodate more aircraft communications. However, this solution alone is insufficient for the growing demands of modern air traffic. Despite its limitations, VHF remains vital to aviation communication, particularly in scenarios requiring urgent and direct contact between ATC and pilots [1].

5) Controller Pilot Data Link Communications (CPDLC): CPDLC is a data communication system designed to enhance the efficiency of ATC by supplementing traditional voice communication, particularly for non-urgent, routine exchanges between pilots and controllers. CPDLC operates as part of the Future Air Navigation System (FANS), which aims to modernize air traffic management by improving communication, navigation, and surveillance systems.

The primary goal of CPDLC is to reduce congestion on VHF channels by transferring routine communications to a data link. CPDLC messages include instructions such as altitude changes, route clearances, speed adjustments, and other non-urgent commands. These messages are transmitted digitally and displayed on the pilot's visual monitor in the cockpit, ensuring clear and concise communication without the possibility of misinterpretation with voice transmissions.

Technically, CPDLC operates through various communication links, including VHF Data Link Mode 2 (VDL Mode

2), SATCOM, and HF Data Link (HFDL), depending on the airspace. VDL Mode 2 operates in the 118.00 to 136.975 MHz range, similar to traditional VHF voice communication, but it is dedicated to data transmission, reducing congestion on voice channels. This frequency range allows CPDLC to support efficient data exchange, particularly in congested airspaces, and to complement voice communication in areas with high traffic density.

CPDLC is particularly beneficial in oceanic or remote airspace where VHF voice communication may be unavailable, offering seamless communication through satellite links. As CPDLC is globally adopted, it plays a key role in reducing voice communication workload, improving efficiency, and minimizing communication errors [1].

6) Aeronautical Mobile Airport Communications System (AeroMACS): AeroMACS is a digital communication system that provides wireless broadband services at airports, primarily for air-to-ground communications. It operates in the protected and licensed aviation C-band frequency range of 5091 MHz to 5150 MHz. Based on the IEEE 802.16 WiMAX technology, AeroMACS supports both safety-related Air Traffic Services (ATS) and non-safety-related Aeronautical Operational Control (AOC) services. AeroMACS is specifically deployed at airports for ground-based communications, offering data rates between 1.8 Mbps and 9.2 Mbps depending on the modulation schemes, which include adaptive QPSK to 64-QAM. As part of ICAO's Global Air Navigation Plan (GANP), AeroMACS aims to improve communication infrastructure at airports, and its cybersecurity features leverage the WiMAX security framework, utilizing Public Key Infrastructure (PKI) for authentication and data protection [2].

7) Satellite Communication (SATCOM): SATCOM is used for aeronautical communications, especially in remote and oceanic areas where terrestrial communication is not viable. SATCOM services, such as those provided by Inmarsat and Iridium, support air-to-ground and ground-to-air communication. Inmarsat operates in the L-band (1525–1559 MHz and 1626.5–1660.5 MHz) and offers services like Inmarsat Aero and SwiftBroadband with data rates up to 432 kbps. The Iridium Certus system, a second-generation service, provides data rates from 22 kbps to 704 kbps using scheduled access methods. SATCOM systems are integral to air traffic management (ATM), particularly in areas beyond the reach of ground-based systems, providing essential services like ATC voice, data services, and Automatic Dependent Surveillance-Contract (ADS-C). SATCOM operates under the standards of ICAO and RTCA for reliable global aeronautical communication [2].

8) L-band Digital Aeronautical Communications System (LDACS): The L-band Digital Aeronautical Communications System (LDACS) is a ground-based communication system designed for use in continental airspace. It is intended to replace and augment the current VHF Data Link Mode 2 (VDLm2) system by providing much higher data rates, ranging from 0.6 to 2.8 Mbps. LDACS operates in the L-band, specifically between 1110–1156 MHz (for forward link) and 964–1010 MHz (for reverse link). LDACS supports various services, including ATS and AOC, and can handle future applications such as 4D trajectories and secure Ground-Based Augmentation System (GBAS) data transmission. It is currently under standardization by ICAO, focusing on strong cybersecurity measures, though the design and standardization process for LDACS is ongoing [2].

B. Host Identity Protocol (HIP)

The *Host Identity Protocol (HIP)* is a security protocol that operates at the network layer. It is designed to differentiate the dual functions of IP addresses by separating host identification from location addressing. Traditionally, IP addresses serve both as identifiers and locators for devices in a network, which introduces challenges, especially in mobile or multi-homing environments. HIP addresses this by introducing a new namespace, which assigns each host a *Host Identity (HI)*, represented by a cryptographic public key. The IP addresses are then used purely for routing, allowing HIP to support more dynamic and secure communications between devices over the internet [3].

In HIP, each host is associated with one or more IP addresses, which may change as the host's location or network conditions evolve. This allows seamless mobility and multi-homing, even in environments with frequent network changes, as the identity (HI) remains consistent. The protocol supports both IPv4 and IPv6 networks, making it suitable for modern network environments.

1) Host Identity Protocol Base Exchange (BEX): The *HIP Base Exchange (BEX)* is a four-way handshake process that establishes a secure association between two hosts. During BEX, the hosts authenticate each other using their Host Identities (HIs), negotiate cryptographic keys, and set up IPsec security associations for encrypted communication [4]. This exchange provides the foundation for secure communication and supports features like mobility and multihoming.

The process starts with the initiator sending an *I1 message*, initiating communication. The responder replies with an *R1 message*, which includes a cryptographic puzzle, a Diffie-Hellman key (DH), and a signature (SIG) [3]. The cryptographic puzzle helps mitigate Denial-of-Service (DoS) attacks by ensuring that only legitimate connections proceed. The initiator then solves the puzzle and sends the solution, along with its DH key and signature, in the *I2 message*. Finally, the responder verifies the puzzle solution, completes the Diffie-Hellman key exchange, and sends the *R2 message* to confirm the establishment of the secure session (Figure 2) [4].

After the BEX process, HIP uses the *Encapsulating Security Payload (ESP)* from IPsec to encrypt and authenticate all data packets exchanged between the hosts. This guarantees confidentiality, integrity, and authenticity of communication [3].

2) Security Features of HIP: HIP significantly enhances network security through several mechanisms. One of its primary security features is the use of *cryptographic identities* (HIs), which are authenticated during the HIP Base Exchange process. This ensures that only authorized hosts can establish

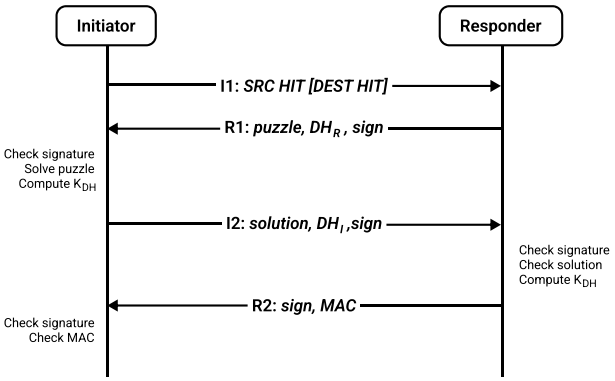

Fig. 2. HIP Base Exchange message procedure.

connections. Moreover, the integration of ESP provides end-to-end encryption for all communication, protecting data from eavesdropping or tampering.

Another important feature is HIP's resilience to *Denial-of-Service (DoS) attacks*. The cryptographic puzzle included in the R1 message helps prevent malicious hosts from overwhelming a server with connection requests. This mechanism ensures that only hosts willing to commit computational resources can establish a session.

HIP's design also supports IPv4 and IPv6 environments, improving its adaptability to modern networking infrastructures. It also supports *mobility and multihoming*, allowing hosts to seamlessly switch networks without losing their session. This is particularly important in scenarios involving mobile devices that change networks frequently (e.g., from Wi-Fi to cellular) [3].

3) Host Identity Tag (HIT): A *Host Identity Tag (HIT)* is a 128-bit cryptographic identifier derived from a hash of the Host Identity (HI). HITs are designed to serve as compact representations of HIs for use in network communications. They provide higher security and identity verification than traditional IP addresses. The fixed length of the HIT also simplifies protocol coding and reduces the cost of managing packet size [3].

4) Hierarchical Host Identity Tag (HHIT): The *Hierarchical Host Identity Tag (HHIT)* builds on the standard HIT by introducing a hierarchical structure for administrative control. HHITs retain the same 128-bit format but allocate 32 bits for hierarchical administration and cryptographic algorithm identifiers. This design optimizes HIT management and enhances security by specifying compatible cryptographic methods between communicating parties [3].

5) Cryptographic Algorithms in HIP: HIP uses various cryptographic algorithms to secure communication and authenticate hosts. These include:

- **Diffie-Hellman Key Exchange:** Used to securely establish a shared secret over an unsecured communication channel during the HIP Base Exchange [4].
- **RSA and ECC:** HIP supports both RSA and *Elliptic Curve Cryptography (ECC)* for generating and verifying signatures. ECC is particularly advantageous due to its smaller key size, which provides equivalent security to RSA but with faster performance [5].
- **Symmetric Algorithms:** For efficient data encryption and decryption, HIP uses symmetric algorithms like *Advanced Encryption Standard (AES)*. AES operates on fixed block sizes with key lengths of 128, 192, or 256 bits, providing high-speed encryption with robust security [6].

6) How HIP Enhances Security and Mobility: HIP significantly enhances both *security* and *mobility* by decoupling host identity from network location. Cryptographic identities (HIs) and the HIP Base Exchange ensure that only legitimate hosts can communicate securely. Once the secure session is established, ESP guarantees that all communication is encrypted and protected from eavesdropping or tampering.

In terms of mobility, HIP allows hosts to seamlessly switch between different networks without disrupting active communication sessions. This makes HIP particularly useful in mobile environments, where devices frequently change network connections (e.g., between Wi-Fi and cellular networks). Similarly, HIP supports *multihoming*, allowing hosts to maintain multiple active network interfaces for redundancy and load balancing.

C. How HIP is Better than LISP

While both HIP and *LISP (Locator/Identifier Separation Protocol)* aim to solve the problem of separating identity from location in networking, HIP offers several advantages, especially in terms of security. *HIP* uses *cryptographic identities* (public keys) to ensure that the communication between hosts is secure and authenticated. In contrast, *LISP* focuses primarily on improving routing scalability by introducing a mapping system for locators (RLOCs) and identifiers (EIDs). HIP's use of cryptography inherently provides *stronger security guarantees*, including *end-to-end encryption* and *protection from spoofing attacks*.

In contrast, *LISP* does not inherently provide these security features; it focuses more on *network efficiency and routing optimization*. HIP also supports *mobility* more effectively than LISP. HIP allows devices to move between networks without losing identity or disrupting ongoing sessions, making it a better choice for mobile environments. On the other hand, LISP primarily addresses routing issues and is less concerned with end-to-end security and seamless mobility.

Overall, HIP offers a *more comprehensive* solution for scenarios requiring both *security and mobility*. At the same time, LISP excels in optimizing network routing but lacks the security features and mobility support provided by HIP.

D. NS-3

Network Simulator 3 (NS-3) is a discrete-event network simulator designed to model and simulate advanced network scenarios [7]. It supports various technologies, including Wi-Fi, WiMAX, and more. The goal of the NS-3 project is to build a well-documented and robust simulation core to be used for networking research.

E. CORE

The Common Open Research Emulator (CORE) is a network emulator that is mainly used for network and protocol research [8]. CORE provides an environment for running real applications on a Linux platform. CORE also provides a drag-and-drop GUI for easy configuration and setup.

F. Simulation Parameters

The simulation parameters will be the same as those used in the report *SAPIENT: Enabling Real-Time Monitoring and Control in the Future Communication Infrastructure of Air Traffic Management* [9]. The simulation parameters are listed in Table I. A communication network covering an area of 2400 x 2400 km^2 will be simulated with 4 satellites and 16 LDACS antennas.

TABLE I
MAIN SIMULATION PARAMETERS

Parameter Name	**Value**
Simulation duration	24 hours
Warm-up duration	100 seconds
# of replicas	6
# ACs	80 to 200
Flying speed	900 km/h
Low-Priority App	
- Packet size	100 Bytes
- Inter-packet time	1 second
High-Priority App	
- Packet size	40 Bytes
- Inter-packet time	40 milliseconds
Transport layer	
- Protocol	User Datagram Protocol (UDP)
TERRESTRIAL DL	
- Slot duration	60 milliseconds
- Slot size FW	2236.416 Bytes (291.2 kbit/s)
- Slot size RT	1691.904 Bytes (220 kbit/s)
SATCOM DL	
- Slot duration	224 milliseconds
- Slot size FW	12000 Bytes (420 kbit/s)
- Slot size RT	1800 Bytes (60 kbit/s)

IV. METHOD

In this section, a walkthrough of the resources available during the project will be presented, followed by a thorough explanation of what the simulation environment should contain to successfully reflect a realistic FCI simulation scenario. Additionally, an alternative approach using CORE will be presented.

A. HIP & Tap-Bridge

During the initialization phase of the project, the choice of simulator was the first major decision. After research and active communication with experts in the field, the choice between two implementations of HIP became clear. The differences between them lay in the programming language they were written in and the functionalities related to their configuration, such as connection to and from tap interfaces and routing to and from the Docker container.

The first version, written in the programming language C, was the oldest and lacked documentation, as the websites hosting the documentation were deprecated. Additionally, it was not well-maintained and had compatibility issues with newer systems and dependencies.

To address this, the second version, written in Python, was analyzed for compatibility with the project's needs. Mapping its documentation revealed that the Python version was better maintained and ran without errors on modern systems.

After evaluating NS-3 and the integration possibilities with HIP, it became apparent that there was no native implementation of HIP in the simulator. Additionally, the HIP source code was designed to run as network daemons rather than within the simulator. Therefore, the following approach was proposed:

1) Spin up a Docker image with a running HIP network daemon.
2) Create a virtual Ethernet device, where one end is attached to the Docker container and the other end to the host, using Linux Kernel namespacing.
3) Create a TAP device in promiscuous mode to listen to all incoming packets.
4) Create an OS bridge, attaching one end to the virtual Ethernet device on the host and the other to the TAP device.
5) Write an NS-3 script that uses the Tap-Bridge class to connect to the TAP device and an NS-3 net device via Inter-Process Communication.

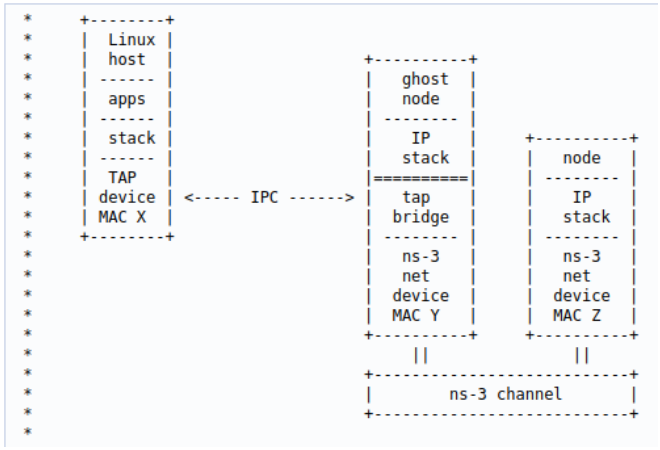

Fig. 3. Tap-Bridge using the UseLocal mode

By utilizing this method, it becomes possible to create multiple HIP nodes, such as aircraft, LDACS ground stations, AeroMACS ATCs, and SATCOM satellites, without needing to develop a custom HIP implementation. This method exposes the HIP nodes to NS-3, enabling realistic simulation of the FCI infrastructure.

B. NS-3 Simulation & Components

To properly simulate each communication system (LDACS, SATCOM, and AeroMACS), the NS-3 network simulator was used to model their respective parameters and behaviors. Suitable modules were chosen to represent each technology since no native modules for LDACS, SATCOM, or AeroMACS

were available in NS-3. After evaluating various options, the following modules were selected:

1) LDACS: To simulate the LDACS component, the project utilized the IEEE 802.11a (Wi-Fi) standard. This ensured realistic simulation parameters such as channel bandwidth, modulation schemes, and error correction techniques. Operating in the 5 GHz frequency band, 802.11a supports data rates up to 54 Mbps and uses Orthogonal Frequency Division Multiplexing (OFDM) for data transmission. Add-ons such as handover support and packet tracing were included to enhance the module's capabilities and measurement accuracy.

2) AeroMACS: The WiMAX module in NS-3 was used to simulate AeroMACS, as it adheres to the IEEE 802.16 standard, which forms the foundation of AeroMACS. This allowed accurate representation of key functionalities and protocols. The module supports creating subscriber stations (SS) and base stations (BS) for effective wireless communication modeling. However, inter-domain handover between base stations was not supported, limiting seamless transitions during simulations.

3) SATCOM: Since NS-3 lacked a dedicated satellite module, an extension called Satellite Network Simulator 3 (SNS3) was used for SATCOM simulation. Although the latest SNS3 release was incompatible with NS-3.42, its development branch provided a working implementation. Given SNS3's additional dependencies, setup scripts were updated to automate its download and installation rather than including it directly in the project repository.

C. Alternative Approach - CORE

Towards the end of the project, an alternative approach using CORE emulator was explored to obtain measurable results. Simulations were conducted for each technology (LDACS, AeroMACS, and SATCOM) to measure the baseline key exchange time. The simulation parameters for each technology are provided in Tables II, III, and IV. The chosen parameters were selected to accurately reflect real-world operational conditions and constraints for each technology. All simulations were conducted using Ubuntu 22.04 running as a VirtualBox within a Windows machine. The same scenario, depicted in Figure 4, was used for all three simulations. Measurements were obtained by pinging machine n1 from machine n4.

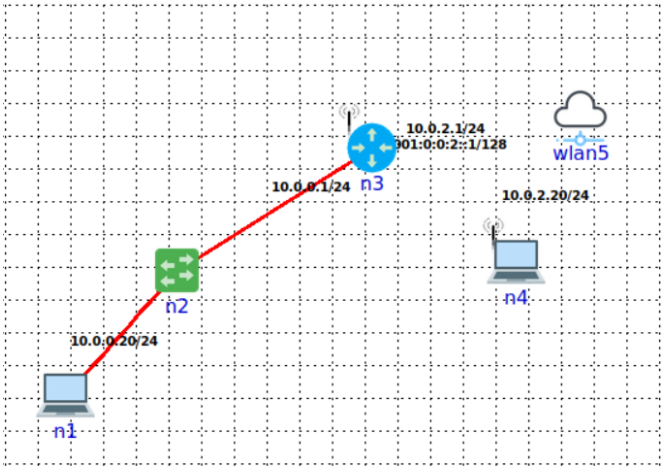

Fig. 4. Simulation Scenario in CORE

TABLE II
LDACS SIMULATION PARAMETERS

Parameter Name	Value
Bandwidth (bps)	1000000
Transmission delay (usec)	100
Loss (%)	0.1
Transmission jitter (usec)	100
Promiscuous mode	off
Wireless range (pixels)	275

TABLE III
AEROMACS SIMULATION PARAMETERS

Parameter Name	Value
Bandwidth (bps)	20000000
Transmission delay (usec)	250
Loss (%)	0.1
Transmission jitter (usec)	100
Promiscuous mode	off
Wireless range (pixels)	275

TABLE IV
SATCOM SIMULATION PARAMETERS

Parameter Name	Value
Bandwidth (bps)	20000000
Transmission delay (usec)	250000
Loss (%)	1.0
Transmission jitter (usec)	100
Promiscuous mode	off
Wireless range (pixels)	27500

V. RESULT

At the current state of the project, as it reaches the handover deadline, certain limitations in the results are apparent. In this section, an overview of the implemented simulation modules is presented, followed by a walkthrough of the Docker container hosting the HIP daemon. Furthermore, a detailed presentation of the project's limitations provides insights into where the results failed to deliver a functional simulation environment.

Due to limitations in the project results, the project goals remain unfulfilled. However, all project goals have been partially achieved. The lack of a functioning simulation environment makes it impossible to generate performance indexes relevant to the project goals **Seamless Handover** and **Multi-Homing Support**. Detailed explanations regarding these limitations and the reasons for the environment's shortcomings are covered in subsections addressing respective components later in this section.

Nevertheless, the lack of a fully functional simulation environment does not imply a complete absence of results. The project demonstrates the implementation of key components that, with additional time and a greater focus on integration, have the potential to meet the defined goals. Notably, the setup of HIP using Docker and the *Tap-Bridge* functionality is a success. The project presents a working simulation where secure communication is maintained between two nodes using HIP and *Tap-Bridge* in NS-3. Further details are provided in Section V-A4.

As such, while the project failed to fully deliver the promised goals defined at the project's inception, the results partially fulfill the objectives with suboptimal solutions.

A. NS-3

To accurately simulate the FCI components presented in Section III, each component has been implemented and tested in the NS-3 simulation environment to the extent that they correspond closely to their realistic counterparts within a geographically restrictive area of 2400 x 2400 km^2.

At the current state of the project, no published LDACS, AeroMACS, or SATCOM modules for NS-3 are functional and ready for integration. To address this, simulation environments and core functionalities have been set up to enable HIP simulations in NS-3. The following subsections cover each component's integration and details about their implementation.

1) LDACS: The simulation is configured with multiple ground stations aligned horizontally, with realistic distances between them, and one aircraft following a trajectory above these ground stations. As the aircraft moves, it performs handovers with ground stations whenever a new ground station becomes the closest. After each handover, the aircraft begins rejecting packets sent from the previously connected ground station, simulating a realistic velocity and behavior.

The primary limitation at the end of the project is the inability to test the LDACS component in a simulation environment with a larger set of diverse components. A realistic scenario covering a geographical area equivalent to a small country or a larger region with multiple airports, satellites, ground stations, and aircraft remains unexplored.

2) AeroMACS: In the current setup, two subscriber stations and a base station are configured to represent an airport scenario. This simulates a rolling aircraft moving at a speed of 10 m/s while maintaining wireless communication with Air Traffic Control. Unfortunately, when the aircraft moves too far from the base station, it repeatedly attempts to re-register, which eventually causes the simulation to crash.

3) SATCOM: With the help of SNS3, communication between an airplane and a ground station through a satellite is simulated. Parameters for SATCOM, detailed in Table I, were used in the simulation, with the addition of a delay based on the use of Low Earth Orbit satellites.

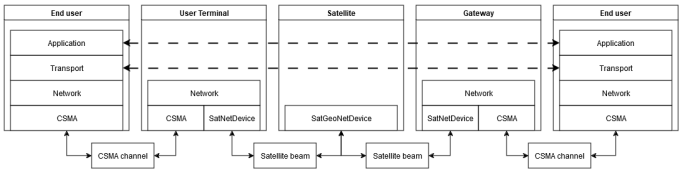

Fig. 5. SNS3 Architecture

Both the airplane and ground station consist of two nodes: one End User together with one User Terminal (UT) for airplanes, and one Gateway (GW) for ground stations. The pair of nodes communicate through a Carrier Sense Multiple Access (CSMA) channel with chosen parameters set. Packets are then sent through a satellite beam between the User Terminal and Gateway.

4) HIP and Docker: The developed HIP configuration has been set up using Docker. A default node has been configured with a running instance of HIP (python version, called *Cutehip*), tap-bridge functionality, and configured network interfaces to enable external traffic communication using HIP. This default node is later scaled up to match the needs of the simulation environment. Further, these nodes are connected with the NS-3 interface in order to send and receive packets through the simulation environment.

At the time of project finalization, a bug in the *Cutehip* source code had been found that seemed to create a hard upper limit of the amounts of packets that could be transferred. This hard limit was after multiple test runs defined to be 15 packets. Due to this, and the amount of packets needed for HIP to successfully generate the handshake (BEX), a realistic simulation using HIP is simply not possible.

B. CORE

During the last weeks of the project, the problems regarding the integration of HIP and NS-3 unfortunately halted progress. To counter this, the alternative approach presented in Section IV-C was implemented to obtain measurable results. Here, the Base Exchange (BEX) times were in focus to accurately measure the delays associated with using HIP for multi-origin communication.

1) LDACS: The results of the LDACS simulation are presented in Table V. Without using HIP, and with the simulation parameters configured for LDACS, the average time to receive the first ping response was 4.284 ms. When using HIP and pinging the LSI of the other node, the average time to receive the first response (including BEX) increased to 1044 ms. This indicates an additional time cost of approximately 1040 ms on average.

TABLE V
PING TIMES IN MILLISECONDS WITH LDACS

Scenario	Time (ms)
Pinging IP	
	4.27
	3.21
	4.55
	3.55
	5.84
Average	**4.284**
Ping LSI with BEX	
	1029
	1039
	1064
	1045
	1043
Average	**1044**

2) AeroMACS: The results of the AeroMACS simulation are presented in Table VI. Without using HIP, and with the simulation parameters configured for AeroMACS, the average time to receive the first ping response was 2.628 ms. When using HIP and pinging the LSI of the other node, the average time to receive the first response (including BEX) increased to 1034.4 ms. This indicates an additional time cost of approximately 1032 ms on average.

TABLE VI
PING TIMES IN MILLISECONDS WITH AEROMACS

Scenario	Time (ms)
Pinging IP	
	1.51
	6.61
	1.88
	1.06
	2.08
Average	**2.628**
Ping LSI with BEX	
	1069
	1026
	1031
	1024
	1022
Average	**1034.4**

3) SATCOM: The results of the SATCOM simulation are presented in Table VII. Without using HIP, and with the simulation parameters configured for SATCOM, the average time to receive the first ping response was 571.6 ms. When using HIP and pinging the LSI of the other node, the average time to receive the first response (including BEX) increased to 3377.6 ms. This indicates an additional time cost of approximately 2806 ms on average.

TABLE VII
PING TIMES IN MILLISECONDS WITH SATCOM

Scenario	Time (ms)
Pinging IP	
	508
	611
	573
	612
	554
Average	**571.6**
Ping LSI with BEX	
	3074
	3346
	3147
	4167
	3154
Average	**3377.6**

VI. DISCUSSION

The results of this project are limited when measured against the initial project goals. The project focused on implementing a relatively large-scale simulation environment, but due to the limitations mentioned in Section V and a lack of time during the integration phase, the project cannot be considered complete.

The project results represent partial components that, collectively, have the potential to achieve the goals. This project explored NS-3's capacity to simulate network scenarios using different technologies and set up a solid foundation for testing and simulating HIP within the context of FCI.

Additionally, the project explored an experimental approach to running HIP using the CORE emulator. This is discussed further in Section VI-A.

A. CORE

Due to integration problems with HIP and NS-3 encountered during the project's final weeks, the alternative approach presents promising results. As covered in Section V-B, we were able to measure the delay for base key exchange across the different simulated technologies. Building on this, more advanced network scenarios can be designed to explore the performance and challenges of using HIP within the FCI.

B. Challenges and Limitations

As presented in the results chapter, several factors limited the outcomes of this project.

When executing the method mentioned in Section IV, we discovered a buffer-related bug when NS-3 tried to decode an ARP packet, which caused the project to come to a halt.

VII. FUTURE WORK

This chapter discusses subsequent tasks that can further develop and enhance the work achieved in this project.

A. Integration

To build upon the progress achieved in this project, the primary focus should be on integrating the various components into a single simulation. As outlined in the results chapter, the individual parts of the project were developed and tested independently but have not yet been unified. Future efforts should prioritize combining these standalone components, ensuring smooth interaction between them, and validating the performance of the complete simulation. This integration would enable a more thorough evaluation of the project's overall goals.

B. Enhancing Network Scenarios for Testing HIP within FCI

To test more advanced scenarios, larger and more complex setups could be implemented to demonstrate HIP's capabilities within the FCI context. The scenarios created in this project were relatively limited and small. To thoroughly evaluate seamless handover, multi-homing support, and secure communication capabilities, more advanced testing scenarios are essential.

C. Improved Handover Functionality for LDACS in NS-3

As mentioned in Section V-A1, to properly simulate LDACS functionality in a realistic FCI environment, communication with multiple aircraft and satellites is crucial. The handover created in the LDACS simulation works but could be improved.

D. Interdomain Handover for WiMAX Module in NS-3

To properly simulate handovers for moving subscriber stations between base stations, implementing this feature in NS-3 is essential. This allows for scenarios where subscriber stations dynamically switch connections between base stations, ensuring uninterrupted communication. It also enables evaluation of network performance under mobility conditions. If this is not feasible, exploring an alternative module may be necessary. An alternative solution could involve utilizing a different module in NS-3 that supports interdomain handover capabilities.

References

[1] Suleman Khan, Andrei Gurtov, An Breaken, et al. "A security model for controller-pilot data communication link". In: *2021 Integrated Communications Navigation and Surveillance Conference (ICNS)*. IEEE. 2021, pp. 1–10.

[2] Nils Mäurer, Tobias Guggemos, Thomas Ewert, et al. "Security in Digital Aeronautical Communications: A Comprehensive Gap Analysis". In: *International Journal of Critical Infrastructure Protection* 38 (2022), p. 100549.

[3] *RFC 7401 - Host Identity Protocol Version 2 (HIPv2)*. https://tools.ietf.org/html/rfc7401. Accessed: 2024-10-13. 2015.

[4] W. Diffie and M. E. Hellman. "New Directions in Cryptography". In: *IEEE Transactions on Information Theory* 22.6 (1976), pp. 644–654. DOI: 10.1109/TIT.1976.1055638.

[5] V. S. Miller. "Use of Elliptic Curves in Cryptography". In: *Advances in Cryptology — CRYPTO '85 Proceedings*. Ed. by H. C. Williams. Springer, 1985, pp. 417–426. DOI: 10.1007/3-540-39799-X_31.

[6] National Institute of Standards and Technology. *FIPS PUB 197: Advanced Encryption Standard*. https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197.pdf. Accessed: 2024-10-13. 2001.

[7] Nsnam. *About*. https://www.nsnam.org/about/. Accessed: 2024-11-29.

[8] *CORE Documentation*. https://coreemu.github.io/core/index.html. Accessed: 2024-12-14.

[9] Antonio Virdis, Giovanni Stea, and Gianluca Dini. *SAPIENT: Enabling Real-Time Monitoring and Control in the Future Communication Infrastructure of Air Traffic Management*. https://ieeexplore.ieee.org/document/9062565. Aug. 2021.