international journal of critical infrastructure protection 42 (2023) 100619

Contents lists available at ScienceDirect

International Journal of Critical Infrastructure Protection

journal homepage: www.elsevier.com/locate/ijcip

AKAASH: A realizable authentication, key agreement, and secure handover approach for controller-pilot data link communications

Suleman Khan [a], Gurjot Singh Gaba [a,*], An Braeken [b], Pardeep Kumar [c], Andrei Gurtov [a]

[a] IDA - Department of Computer and Information Science, Linköping University, Sweden
[b] Industrial Engineering Department (INDI), Vrije Universiteit Brussel (VUB), Belgium
[c] Swansea University, Swansea, UK

ARTICLE INFO

Keywords:
Authentication
CPDLC
Critical infrastructure
Safety
Security

ABSTRACT

Controller-Pilot Data Link Communications (CPDLC) are rapidly replacing voice-based Air Traffic Control (ATC) communications worldwide. Being digital, CPDLC is highly resilient and bandwidth efficient, which makes it the best choice for traffic-congested airports. Although CPDLC initially seems to be a perfect solution for modern-day ATC operations, it suffers from serious security issues. For instance, eavesdropping, spoofing, man-in-the-middle, message replay, impersonation attacks, etc. Cyber attacks on the aviation communication network could be hazardous, leading to fatal aircraft incidents and causing damage to individuals, service providers, and the aviation industry. Therefore, we propose a new security model called AKAASH, enabling several paramount security services, such as efficient and robust mutual authentication, key establishment, and a secure handover approach for the CPDLC-enabled aviation communication network. We implement the approach on hardware to examine the practicality of the proposed approach and verify its computational and communication efficiency and efficacy. We investigate the robustness of AKAASH through formal (proverif) and informal security analysis. The analysis reveals that the AKAASH adheres to the CPDLC standards and can easily integrate into the CPDLC framework.

1. Introduction

With about 4.3 billion passengers traveling on different airlines each year, civil aviation has become one of the world's most rapidly growing business sectors. Amongst various reasons to fly, tourists are the major contributors to the reasons behind the booming aviation industry. Experts estimate that by 2036, the civil aviation industry will easily breach a market value of USD 5 trillion and will likely employ more than 98 million people [1,2].

In the last few decades, civil aviation as an industry has seen a massive evolution in terms of the adoption of digital technological aids, such as Controller pilot data link communication (CPDLC), Automatic Dependent Surveillance-Broadcast (ADS-B), etc. CPDLC is a digital communication technology designed for aviation and has been in use since 2000. Before this, airplanes and ground stations used analog communication set-ups for most ground and air-to-ground operations. However, the few reasons to replace analog technologies with digital counterparts were the difficulty in storing and interpretation, extensive bandwidth requirements, inefficient compression, and susceptibility to noise and attacks. This transition of communication mode has significantly fulfilled the desired objectives of making air travel safe and reliable [3].

Transportation via airplanes is more sensitive than trains, trams, ferries, and vehicles on roads because any miscommunication or loss of communication between the entities involved (pilots, ATC) can cause severe threats to passengers on board, aircraft crew, infrastructure, etc. [4]. For instance, crash investigations of the Boeing 777 in San Francisco, the Airbus A300 in Birmingham, and the Piper PA-32R - Eurocopter AS350 in New Jersey revealed that interruption in each case occurred in communication was the sole reason for the crash and loss of lives [5]. Unfortunately, despite spending millions of dollars on the modernization of the infrastructure required for the safety of passengers and goods, there are still many safety and security measures to be addressed in civil aviation.

The conventional means of communication between the cockpit crew and the ground station staff in ATC was through very high frequency (VHF) transceivers. The ATC operators used VHF for all critical and non-critical communications, including weather forecasting, travel

* Corresponding author.
E-mail addresses: suleman.khan@liu.se (S. Khan), gurjot.singh@liu.se (G.S. Gaba), an.braeken@vub.ac.be (A. Braeken), pardeep.kumar@swansea.ac.uk (P. Kumar), andrei.gurtov@liu.se (A. Gurtov).

https://doi.org/10.1016/j.ijcip.2023.100619
Received 6 December 2022; Received in revised form 15 June 2023; Accepted 23 June 2023
Available online 28 June 2023

routes, and restricted airspace [6]. VHF communication is prevalent and has been in use for a long time in the aviation sector. However, VHF suffers from three acute issues: (i) delay in sharing critical messages due to its half-duplex nature, (ii) crosstalk due to channel overlapping, and (iii) queuing at ATC leading to increased response time. In response to increased air traffic and the aforementioned issues in VHF communication systems, aviation experts proposed CPDLC as a digital alternative to noisy and unreliable analog modes of communication, primarily for non-critical message exchanges.

In modern aircraft, voice and data links coexist to manage air traffic service (ATS) communications. CPDLC is utilized mainly for non-time-critical conversations, including level assignments, crossing constraints, lateral deviations, route adjustments, clearances, speed assignments, radio frequency assignments, etc. In Europe, CPDLC uses the Aeronautical Telecommunications Network (ATN) with very high-frequency data link mode 2 (VDL2, 118 to 136.975 MHz) at a data rate of 31.5 kilobits per sec. [7]. Whereas, in Australia and USA, CPDLC uses the satellite communications-based Future Air Navigation System (FANS-1/A) to facilitate air-ground communications. Although ATN and FANS-1/A are compatible, attempts to converge them over time to ensure uniformity are underway [8,9].

Although CPDLC has merits from an easy communication perspective, it fails to prevent cyber abuses, which can often be fatal for the entire system [10]. Because the exchange of messages in CPDLC systems occurs in plaintext over an unsecured wireless medium, exploitation by cyber attackers is fairly simple. For instance, with minimal hardware, such as an SDR dongle, VHF airband antenna, and amplifier, an attacker can overhear the CPDLC communication and alter messages. Additionally, the attacker can inject messages and impersonate the identities to obtain illegitimate control. These attacks can lead to catastrophic consequences that could directly impact the confidentiality, integrity, and availability of the communication services and, eventually, the entire ecosystem [11].

The European Air Traffic Management-Computer Emergency Response Team (EATM-CERT), in its report, highlighted that about 775 critical cyber-attacks on airlines were reported in 2020 alone. In one of the incidents, hackers took control of LOT airline's ground control systems. They paralyzed their system for many hours, resulting in the cancellation of 10 flights and a delay in over another dozen flights. Apart from being a prime transportation mode for diplomats, politicians, VIPs, and the general public at large, airlines are also an integral part of the food supply chain. Hence, the current limitations and drawbacks in the context of cyber security in CPDLC pose several threats with long-reaching impacts. Therefore, there is an urgent requirement for a bandwidth-efficient, computationally inexpensive, reliable, and robust approach that provides a high level of security and privacy to protect ground-air communication from cyber-attacks.

1.1. Our contributions

To understand and address the security issues in CPDLC, we have made the following contributions:

- We eavesdropped on the communication channel used by the ATC and airplane crews for communication through a Hack-one SDR dongle at a site near Arlanda Airport, Sweden, and captured the CPDLC messages.
- We examined the captured CPDLC messages to understand the message structure and its various components.
- After gaining an in-depth knowledge of the CPDLC communication and message structure, we analyzed and identified the security vulnerabilities in the CPDLC that attackers could exploit.
- We proposed a new security model called AKAASH, enabling several essential security services, such as authentication, key agreement, data integrity, confidentiality, anonymity, and freshness. The AKAASH would also enable a secure handover when an aircraft moves from one zone to another. In addition, AKAASH is safe from various attacks, such as eavesdropping, data tampering, MITM, etc.
- We experimented with the AKAASH on a test bed comprising a raspberry pi (aircraft) and processing units (ground station) and analyzed the computation costs, timings, and payload compatibility to demonstrate the practical applicability of our solution.
- Lastly, we verified the robustness of the proposed solution against various malicious attacks through formal (ProVerif) and informal analysis.

1.2. Paper organization

The organization of the paper is as follows. Section 2 provides a comprehensive literature review, followed by a discussion on motivation and research gaps. Section 3 comprises preliminaries and background, including the system, adversary model, security goals, and data collection. Section 4 elaborates on the proposed mutual authentication, key exchange, and secure handover approach, followed by formal and informal security analysis in Section 5. Section 6 details the experimental setup and discusses the performance of the proposed solution. Finally, the conclusion and future work is discussed in Section 7.

2. Related work

Cyber-security research groups have identified security vulnerabilities in the CPDLC. Some of the proposed cryptography-based solutions to secure the CPDLC framework are discussed in this section. Strohmeier in [12] discusses various possible threats (e.g., jamming, eavesdropping, message injection, message deletion) to ground-air communication systems. Strohmeier did not consider a vast threat landscape while examining the effectiveness of the proposed countermeasure. Moreover, Strohmeier did not suggest any measures to secure the handover between Air Traffic Service Units (ATSUs). Inspired by [12], Eskilsson et al. [13] captured CPDLC messages using a HackRF One RTL-SDR dongle to assess its strength against malicious cyber-attacks. The experiments indicate that CPDLC is prone to uplink and downlink attacks.

Similarly, Gurtov et al. [14] recognized the security limitations and developed a threat model for the CPDLC use case. The authors recommend elliptic curve cryptography (ECC), host-identity protocol (HIP), and identity-defined networking (IDN) as potential countermeasures against cyber-attacks. Because the authors did not implement the suggested solution, the practicability of the solution remains uncertain. Apart from this, Gurtov et al. has not discussed handover-related security concerns and resolutions.

By contrast, the authors in [15] elaborate on the security threats in CPDLC due to the absence of authentication mechanisms. The authors also discuss the requirements set by the International Civil Aviation Organization (ICAO) for secure air-ground communications. The authors suggest the use of fundamental cryptography primitives like encryption and well-known authentication schemes like Diffie–Hellman to safeguard CPDLC against cyber abuses. The advised solutions were neither examined for robustness nor verified for compatibility.

Similarly to, [12–14], Getachew and Griner [16] investigated the security issues concerning the air and ground entities in the CPDLC. They presented a novel two-step authentication process, wherein (1) an ECC-based mechanism is used to authenticate the entities during initial contact, and thereafter, (2) an Aeronautical Telecommunications Network (ATN) keyed message authentication code (MAC) is used to authenticate the CPDLC messages. The authors claim that the scheme is energy efficient due to the use of ECC; however, its efficiency has not been confirmed.

Motivated by Getachew and Griner, Khan et al. [17] expanded the ECC-based mutual authentication scheme to secure the handovers

Table 1
Summary of related work on CPDLC security.

Reference	Highlighted security issues	Suggested countermeasures	Drawbacks
[6]	MITM attacks	PKI-based authentication	Compatibility, computation, and communication costs are not considered.
[12]	Jamming, eavesdropping, message injection, and message deletion attacks	–	The protocol only covers ground-ground handover. Security of air-ground handover was not taken into consideration.
[13]	Up- and down-link attacks	–	Repercussions of session hijacking and denial of service were not considered.
[14]	Eavesdropping, message injection, and message modification attacks	HIP-based authentication	Not taken into account CPDLC compatibility, computation, and communication costs.
[15]	Injection, jamming, and DoS	AES and DHKE based secure key-exchange	Robustness and compatibility not verified.
[16]	Message modification and injection attacks	ECC-based authentication	The protocol is lacking both formal and informal verification, and there is no discussion of its practicability.
[17]	MITM attacks	ECC-based authentication	The protocol only covers air-ground handover and does not address ground-based handover.

Compatibility: The authors did not confirm whether their solution can be practically integrated into the CPDLC.

between the ATSUs. The computational burden and communication delays in the scheme developed by Khan et al. became the roadblock to its adoption. Smailes et al. [6] demonstrated that CPDLC is prone to MITM attacks. They recommended three countermeasures, including PKI-based architecture, to counter MITM attacks without modifying the standard protocols of CPDLC. The proposed solutions, however, were not tested on a testbed to assess their reliability and communication costs. The discussion is restricted to MITM attacks, so the behavior of the recommended solutions in a compromised environment (other than MITM) cannot be predicted.

The researchers [6,12–17] have made substantial efforts to identify and address the security concerns in CPDLC. In most of the cited literature [12–16], the authors attempt to efficiently safeguard the authentication and key establishment mechanisms using ECC. Only Smailes et al. [6] and Khan et al. [17] offered a solution to protect the handovers from cyber-abuses. The summary of the related literature is shown in Table 1.

Since the prevailing solutions were prepared using a very narrow threat landscape and without handover protection, aeronautical security developers, verifiers, and implementers did not find the aforementioned recommendations particularly convincing in developing comprehensive safeguards. In conclusion, the existing schemes' have the following drawbacks: (1) non-compatibility with the CPDLC message structure, (2) absence of formal/informal security analysis, (3) unpredictable behavior, as the schemes' were never implemented, and (4) expensive in terms of compute and delay.

2.1. Problem statement and research motivation

The CPDLC system wirelessly exchanges unencrypted data link messages between the aircraft and ATSUs. CPDLC is susceptible to various cyber-attacks because of the unsecured wireless medium and the absence of security properties. For instance, an adversary can compromise the integrity of CPDLC messages by intercepting and manipulating them using a software-defined radio (SDR) dongle, a VHF airband antenna, and an amplifier. This allows the attacker to impersonate a legitimate ground station or aircraft and transmit fake messages to an aircraft, leading to deviations from the planned route or providing inaccurate information to ATC. Moreover, an attacker can execute a jamming attack, affecting the availability of services at both the aircraft and ground stations, particularly during the handover phase.

Likewise, a masquerading attack can permit the attacker to access classified information and gain illegitimate control of sensitive systems. Additionally, the attacker can eavesdrop on the data traffic without the permission of the communicating parties, impacting confidentiality. Furthermore, the adversary can conduct a MITM attack during the handover phase between different ATSU units, affecting integrity, confidentiality, and authentication.

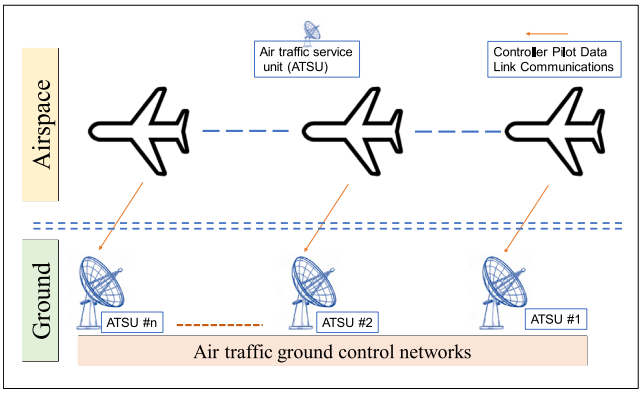

Fig. 1. The ground to airspace communication network model [17].

Such fabricated messages and attacks could be catastrophic for individuals, airplanes, and aviation. For instance, a hacker group allegedly compromised ground-control computers of the state-owned Polish airline LOT, preventing it from issuing flight plans and impacting 1400 passengers [18]. In order to avoid similar cyber abuses in the future, the ATN should possess a high level of security and privacy. Researchers must theoretically and experimentally address the following research questions to ensure the development of a reliable and secure ground-air communication framework for the CPDLC.

(1) How to achieve mutual authentication between aircraft and ATSU with minimum delay and computational cost?
(2) How to prohibit cyber abuses during aircraft-ATSU handovers?
(3) How to examine the practicality and efficacy of the proposed solution?

3. Preliminaries and background

3.1. System model

The safe arrival of an aircraft at its destination is the result of coordinated efforts among three entities involved in ground-air communication. These entities are the aircraft, the ATSU, and data link ground stations (e.g., AeroMACS, VDL Mode 2, LDACS, SATCOM) as shown in Fig. 1. Data links are responsible for facilitating non-critical textual communication between aircraft and ground stations. For instance, a Harris $VDR-2205$ receiver and Harris $VDR-2135$ transmitter pair can be used as a single transceiver to form a datalink via radio [19].

Aircraft use voice (VHF) and data link (CPDLC) technologies to coordinate with ground stations for the safe transportation of passengers,

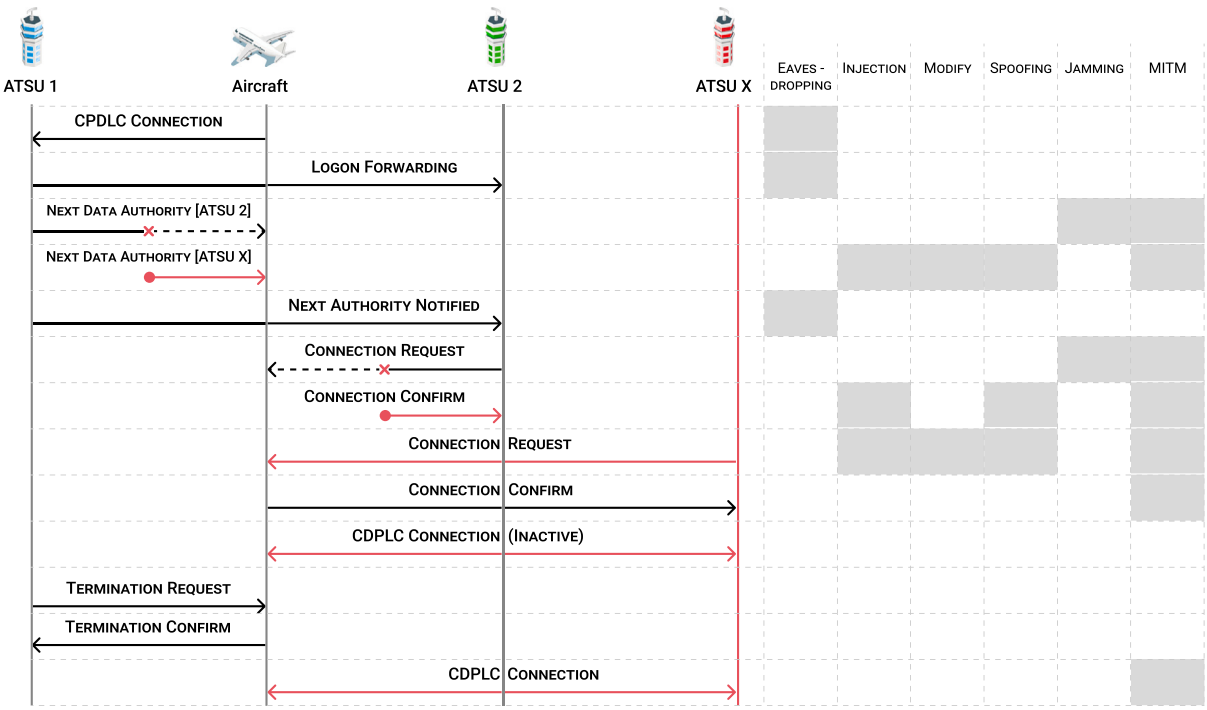

Fig. 2. Possible cyber threats to the CPDLC environment.

goods, and so on. A VHF communication system consists of a radio control panel, transceiver, and an antenna, and it operates between 118.000 MHz to 136.990 MHz. In contrast, data links employ VHF Data Link Mode 2 (VDL-M2) for ground-air message exchanges. The messages are differentially encoded (D8-Phase Shift Keying) before transmission at a data rate of 31.5 kbps over a 25 kHz channel. CPDLC messages are processed and displayed to aircraft and ground station users via a Multi-Function Control and Display Unit (MCDU).

The ATSU's responsibilities include aircraft safety, maintaining a safe distance between aircraft, directing aircraft during takeoff and landing, guiding aircraft crew in harsh weather conditions, and ensuring efficient traffic flow with the smallest number of delays. The ATSU is supported by many types of radar technologies, of which Primary Surveillance Radars (PSR), Secondary Surveillance Radars (SSR), and Mode S are used for air traffic monitoring, while Surface Movement Radars (SMR) are utilized for ground surveillance. Interestingly, ATSU employs the same equipment used by the aircraft for data link communication.

3.2. Adversary model

We have considered Dolev-Yao's (DY) adversarial model as it appropriately describes the adversary's capabilities [20,21]. According to the DY threat model, the cyber attacker can overhear, read, modify, etc., the data shared between different entities over the wireless medium. For instance, a motivated attacker can use publicly available software and hardware to launch a series of attacks to disrupt or take over the CPDLC. Consider a situation where the attacker is interested in performing a man-in-the-middle (MITM) attack. As demonstrated in Fig. 2, the malicious entity can eavesdrop on the communication between aircraft, ATSU 1, and ATSU 2. The hostile entity (ATSU X) can determine the credentials of ATSU 1 from the captured message(s), spoof its identity to appear as honest ATSU 1, and inject the modified message(s) to mislead the aircraft. Next, the malicious entity (ATSU X) intercepts the connection request from ATSU 2 destined for the aircraft, spoofs the aircraft's identity, and confirms the connection request to ATSU 2 on behalf of the aircraft. Likewise, ATSU X can impersonate the identity of ATSU 2 to commence the connection establishment handshake with the legitimate aircraft. As the mimicked entities appear genuine, neither the real aircraft nor ATSU 2 can discover the existence of the MITM attack [6]. These attack(s) can lead to fatal aircraft incidents causing damage to individuals, service providers, and the aviation industry as a whole [22,23].

3.3. Security goals

The following security goals and requirements must be met for a trustworthy and safe CPDLC.

Mutual Authentication and Key Agreement: This security goal ensures that both the aircraft and ATSU authenticate each other's identities and establish a secure communication channel by negotiating a session key before transmitting messages. This is important because it helps prevent masqueraders from sending false or malicious messages, which could lead to unsafe conditions for the aircraft and ATSU, and ensures that messages originate from legitimate sources. The authentication process is typically done using digital certificates or passwords, and the key agreement can be done using key exchange protocols such as Diffie–Hellman.

Integrity: Maintaining integrity is crucial in CPDLC to ensure that messages are not altered during transmission. Protection of the message content, such as flight plan details, altitude assignments, or route changes, is necessary to meet this goal. Any unauthorized changes or modifications to the messages can lead to hazardous outcomes, like landing on the wrong runway. To verify the integrity of messages, secure hash algorithms and digital signatures can be used, which ensures that the CPDLC messages are accurate, complete, and have not been tampered with.

Freshness: The freshness goal ensures that each CPDLC message is unique and not a replay of a previous message. Replay attacks can lead to dangerous outcomes. For example, the fake ATSU (attacker) can deceive the aircraft by replaying level and route change messages. This maneuver would alter the aircraft's height and course, raising safety concerns. Therefore, message freshness is critical to avoid replay

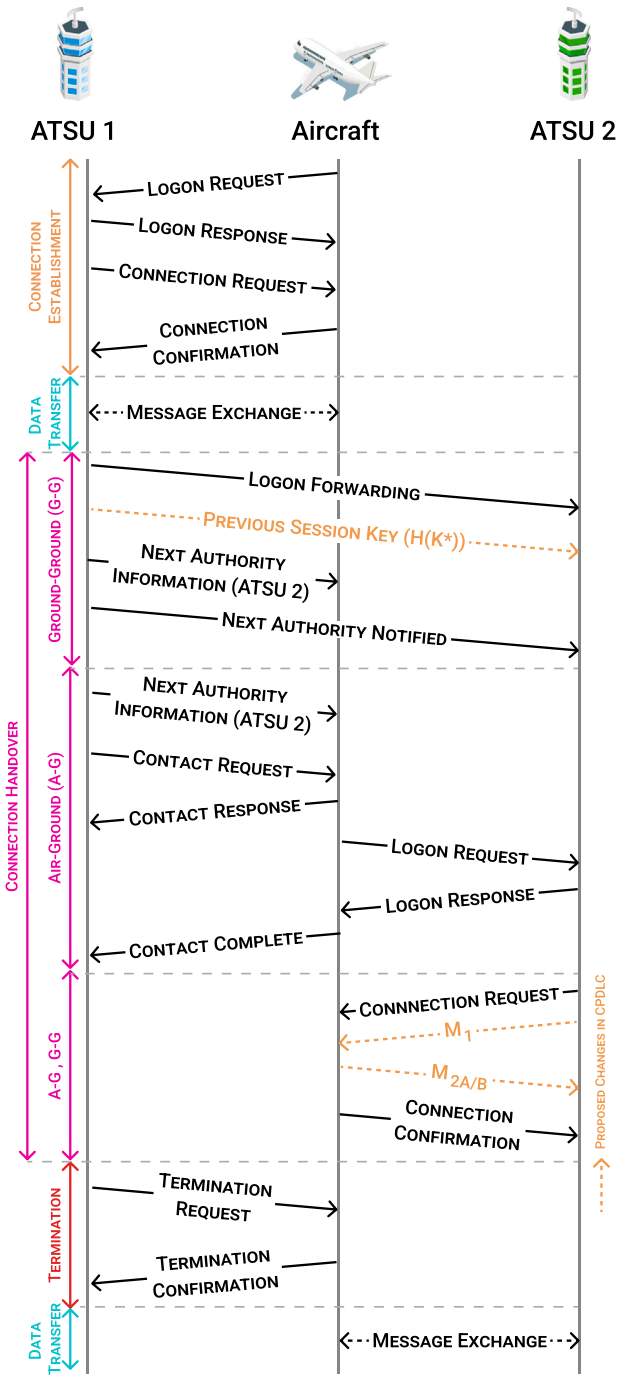

Fig. 3. Connection establishment, data exchange, termination, and proposed secure handover in CPDLC.

attacks, and CPDLC should utilize timestamps or sequence numbers to ensure message freshness.

Confidentiality: Confidentiality ensures that messages exchanged between the aircraft and ATSU are kept confidential and cannot be read by unauthorized entities. This is important because motivated attackers can eavesdrop on the messages and obtain sensitive information, such as lateral deviations, level assignments, speed assignments, and vectoring, leading to potential security risks. Therefore, to protect the information and achieve confidentiality, the system must use encryption and session keys to protect information from unauthorized access.

Anonymity: Anonymity refers to the approach of keeping the identity of an aircraft hidden or anonymous. Maintaining anonymity in CPDLC communication systems is crucial for cyber aviation security. It keeps the aircraft's identity, such as the ICAO address "PIA1234 A", anonymous, thereby protecting sensitive information and reducing the risk of attacks, impersonation, and unauthorized access. By maintaining anonymity, it becomes difficult for attackers to identify and target particular aircraft, which enhances both the safety of the aircraft and the passengers. Anonymity also reduces potential security risks by making it harder for attackers to analyze traffic, further improving CPDLC communication systems' security.

Lightweight: In CPDLC, it is critical to choose the correct cryptography primitives and framework that are lightweight, as complex cryptography computations consume more resources and time, resulting in slower processing and longer round-trip times. This can be particularly dangerous in aviation, where delays in communication could be fatal. Therefore, it is essential to choose lightweight security measures that do not compromise security and processing times performance.

Compatibility: For real-time CPDLC operations, the security protocol must be compatible with the architecture and payload structure of CPDLC. Without the compatibility, the protocol cannot be effectively implemented. Therefore, it is crucial to ensure that the security protocols and CPDLC architecture are aligned to maintain the robustness and working of the protocols.

3.4. Operational flow of CPLDC

Like other stream-oriented communication systems, entities in CPDLC also establish the connection before data transfer. However, CPDLC is not entirely identical to other communication systems as it comprises a handover phase after the data transfer. Following the handover, the aircraft terminates the connection with the previous ATSU and begins communicating with another one. Fig. 3 illustrates the entire concept, and this sub-section provides detailed information on every critical aspect.

Connection Establishment: The aircraft prepares a CPDLC logon request that includes aircraft identifiers, registration, departure, and destination details. Upon receiving the request, ATSU 1 verifies the received information and only accepts genuine requests. Post-verification, ATSU 1 initiates the connection establishment process with the requestor. The connection confirmation from the aircraft completes the handshake and enables the parties to begin sharing the data.

Data Transfer: The CPDLC allows the entities to exchange information, commands, and responses. In general, the ATSU sends commands to the aircraft, while information messages can be transmitted both ways. Typically, the aircraft responds to the received orders with *Wilco* (will comply) or *Unable*. Other messages may need affirmative or negative replies, for instance, *Roger* or *Unable* [6].

Connection Handover: The connection handover is often initiated by the ground station; however, in some cases, it may be triggered by the aircraft. To perform a *ground-to-ground* (G-G) handover, ATSU 1 forwards the logon credentials of the aircraft to ATSU 2 and notifies the aircraft of the subsequent ATSU authority. Upon the final notification from ATSU 1, ATSU 2 generates a connection request and approaches the aircraft for confirmation. The successful confirmation creates a data transfer link between the aircraft and ATSU 2.

While in the *air-to-ground* (A-G) handover, ATSU 1 notifies and asks the aircraft to contact the subsequent ATSU authority. The aircraft acknowledges the request from ATSU 1 and approaches ATSU 2 with its logon credentials. ATSU 2 verifies the authenticity of the credentials and, if they are true, approves the request. Finally, the aircraft informs ATSU 1 regarding its successful logon at ATSU 2 and establishes a data transfer link between itself and ATSU 2.

Connection Termination: After a successful handover, ATSU 1 sends a detachment request to the aircraft. The aircraft accepts the

request and disconnects itself from the ATSU 1. From this point on, the aircraft cooperates with the ATSU 2 until the next handover.

3.5. CPDLC data gathering

We performed the CPDLC message collection experiment to gain a better understanding of the protocol's practical workings and identify any vulnerabilities that may exist in the system. To conduct the experiment, we set up a listening station near Stockholm's Arlanda airport using an RTL-SDR dongle (R820T2), an antenna, and a Chromebook running Crouton. We utilized Tomasz Lemiech's dumpvdl2 software to decode the CPDLC messages and recorded 4040 messages on frequencies of 136.725, 136.775, 136.095, and 136.097 MHz over three days. The decoded plain text of the CPDLC messages contained sensitive information such as ICAO addresses, SSR codes, ATC clearances, and corresponding responses.

Our analysis of CPDLC messages revealed potential vulnerabilities that could compromise the confidentiality and integrity of the information, making it vulnerable to interception and modification by attackers or non-legitimate users. Such a breach could result in severe consequences, including loss of life, damage to property, or disruption of aviation operations. For example, malicious actors could modify an ATC clearance message, causing confusion or misunderstandings between pilots and air traffic control. Similarly, non-legitimate users could exploit sensitive information, such as ICAO addresses or SSR codes, for malicious purposes. Thus, our findings indicate that further security measures, such as robust encryption protocols and improved authentication and authorization mechanisms, are necessary to safeguard the confidentiality and integrity of CPDLC messages.

3.6. Elliptic Curve Cryptography

Elliptic Curve Cryptography (ECC) [20,24,25] offers public key solutions that are more lightweight than RSA. In elliptic curve algebra, $E_{p(a,b)}$ is the curve in the finite field F_p, defined by $y^2 = x^3 + ax + b$, wherein a and b are two constants that satisfy $\Delta = 4a^3 + 27b^2 \neq 0$. G is the base point generator in $E_{p(a,b)}$ of prime order q. The National Institute of Standards and Technology (NIST) organization performs extensive mathematical analysis and testing to determine the most secure and efficient generator base points for cryptographic systems. The EC multiplication is performed as $R = rP = (Rx, Ry)$ with $r \in F_q$ and $R_x, R_y \in F_p$. The strength of ECC depends upon the following factors:

- The Elliptic Curve Discrete Logarithm Problem states that given two EC points R and Q of $E_{p(a,b)}$, it must be computationally challenging to find a parameter $x \in F_q^*$ such that $Q = xR$.
- In the Elliptic Curve Diffie–Hellman Problem, given two EC points, $R = xP, Q = yP$, and two unknown parameters $x, y \in F_q^*$, it must be computationally hard to determine EC point xyP.

3.7. Schnorr signature

The following is the Schnorr signature scheme definition for message M. Assume the sender S possesses the secret key pair (d_S, Q_S) with the private–public key relation $Q_S = d_S G$. Both the receiver and the message verifier have access to the public key Q_S. The sender first chooses a random value $r \in F_q^*$ and computes $R = rG$. Next, it computes $h = H(M, R)$ and the actual signature $s = r - hd_S$. Here, the function H represents a strong hash algorithm resistant against collision, pre-image, and second pre-image attacks. The receiver can verify the signature s, given R, M. The receiver needs to compute $h = H(M, R)$ and check if the equality $sG = R - hQ_S$ holds [26,27].

3.8. Elliptic curve qu vanstone certificates (ECQV)

ECQV-based certificates are used in public key infrastructure (PKI) to associate an owner's identity with their public key. Compared to traditional certificates such as X.509, ECQV-based certificates are much lighter in terms of storage, processing power, and verification speed. This lightweight nature of ECQV-based certificates makes them popular for use in resource constraints and time-sensitive approaches [28,29]. Additionally, ECQV-based certificates offer enhanced security features such as mutual authentication, and they are standardized, making them easier to manage and exchange [26]. In scenarios involving multiple Trusted Third Parties (TTPs), certificate sharing can be complex and pose a higher risk of security breaches. However, in the case of CPDLC, EUROCONTROL is solely responsible for certificate sharing and management, which streamlines the process, making it less complex and more secure. As an intergovernmental organization, EUROCONTROL ensures smooth and secure communication for air traffic management across national borders in numerous European countries.

To generate its key pair in ECQV, the entity must first submit its identifier, ID, and a random EC point, $R_1 = r_1 G$. The CA then selects another point $R_2 = r_2 G$ and computes the certificate $C = R_1 + R_2$. After a certificate is generated, the CA supplies the entity with the auxiliary information $(a = H(ID, C)r_2 + d_{CA})$ needed to determine its private key using the equation $d = a + H(ID, C)r_1$. Furthermore, the entity can derive its public key Q through this equation $Q = dG = H(ID, C)C + Q_{CA}$. Sometimes, the lifetime LT parameter is also included in the certificate to enable self-revocations and prevent replay attacks [30].

4. Mutual authentication, key agreement, and secure handover approach

4.1. General structure

In this section, we illustrate and discuss how the aircraft will securely connect with the ATSUs and establish a new symmetric secret session key (K) during the handover. This session key can then be utilized for communication exchanges as long as the aircraft remains inside the ATSU's operational zone. The use of this key implicitly ensures that both communicating entities are authenticated. As seen in Fig. 4, the scheme consists of four main phases, registration, initialization, mutual authentication and key exchange, and the handover. Table 2 provides the list of notations used in the paper.

4.2. Registration phase

For each geographical region, a trusted third party (TTP) or certificate authority (CA) is responsible for the secure operation of ATSUs and aircraft. For instance, in Europe, this role is played by Eurocontrol.

4.2.1. ATSU registration

The ATSUs receive a longtime certificate from the CA. For compactness and protection against key escrow attacks, we chose to use ECQV certificates. Therefore, each ATSU_i derives its public key by computing, $Q_i = H(ID_i, Cert_i, LT_i)Cert_i + Q_{CAi}$. The certification authority (CA) is a well-known and trusted entity that the ATSUs and aircraft can securely approach by using the following identity and the public key information, ID_{CAi}, Q_{CAi}. The parameter ID_i refers to the identity of the ATSU, $Cert_i$ denotes the certificate generated through the ECQV process, and LT_i represents the termination time of the ECQV certificate. The associated secret key d_i, for which $Q_i = d_i G$ is kept private, is stored in tamper-proof memory. Note that the TTP decides the curve equation, size of the finite field, order of the curve, and base point generator and then shares these parameters with the ATSU and AC. TTP ensures that the curve parameters are well-defined, secure, and suitable for cryptographic applications [27].

Table 2
Notations and description.

Notation	Description	Notation	Description
Random Public: ATSU, AC	R_i, R_a	Encryption, Decryption	E, D
Random Secret: ATSU, AC	r_i, r_a	Certificate Authority	CA
Points on Elliptic Curve	G, R, Q	Subtraction, Addition	+, -
Operator: Comparison, Not Equal	$\equiv$, $\neq$	Hash	H, h
Schnorr Signatures: ATSU, AC	s, s_a	ECQV Certificates: ATSU, AC	$Cert_i$, $Cert_a$
Private Key: ATSU, AC	d_i, d_a	Certificate Lifetime: ATSU, AC	LT_i, LT_a
Public Key: ATSU, CA, AC	Q_i, Q_{CA}, Q_a	Messages: ATSU $\longleftrightarrow$ AC	M_1, M_{2a}, M_{2b}
Secret Session Key	K, K^*	Identity: ATSU, CA, AC	ID_i, ID_{CA}, ID_a
MAKE + SH	Mutual Authentication & Key Exchange + Secure Handover		

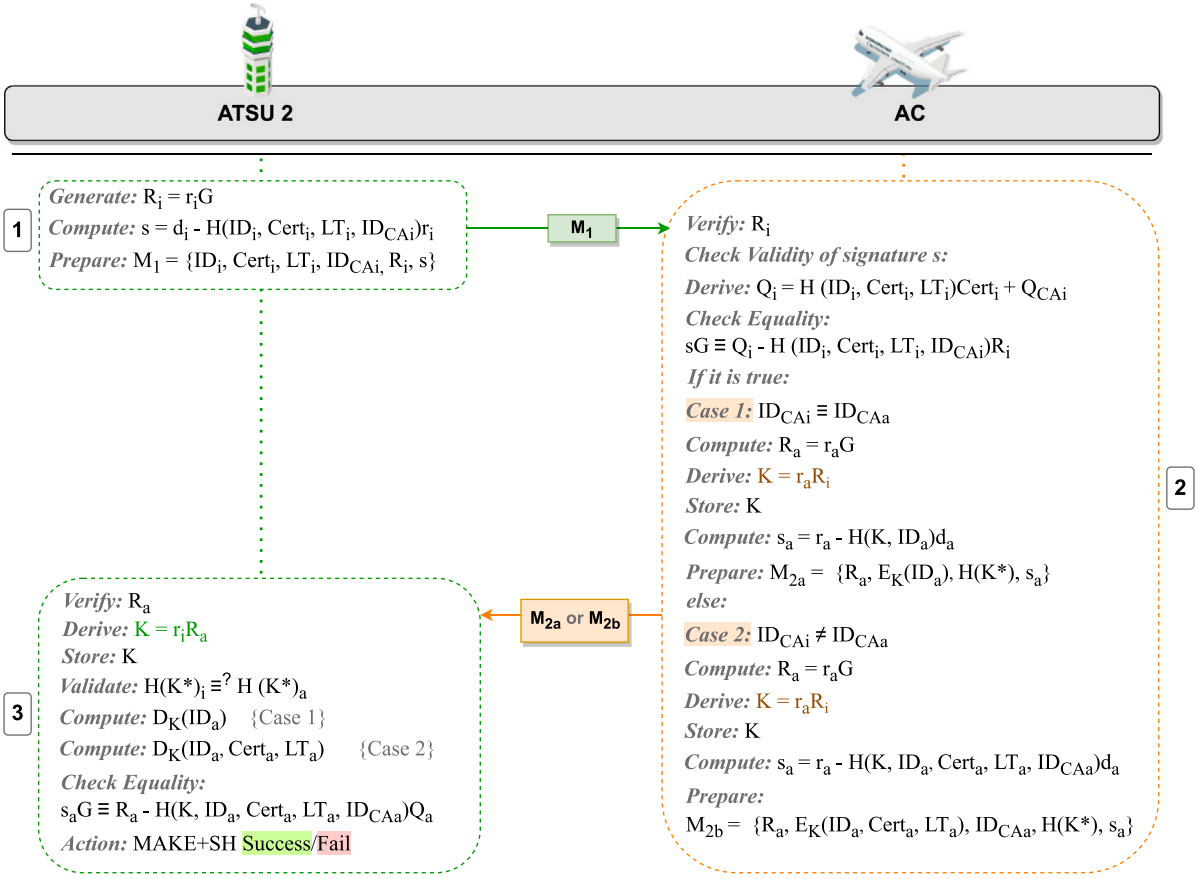

Fig. 4. Proposed mutual authentication, key agreement, and secure handover approach for CPDLC.

4.2.2. Aircraft registration

For each aircraft, the certificate generation is done before the takeoff of each flight to keep potential revocation issues under control. Each aircraft, like ATSU, derives its public key by computing $Q_a = H(ID_a, Cert_a)Cert_a + Q_{CAa}$. ID_{CAa}, Q_{CAa} is the identity and public key of the CA, which is used for proving the authenticity of the aircraft. Note that ID_a contains the identity of the aircraft, together with unique data related to flight destination, departure time, etc.

4.3. Initialization phase

We consider that each ATSU has secure access to the public key repository of its associated CA, allowing it to obtain the public keys of CA-associated aircraft. Additionally, we assume that each ATSU is aware of the public keys of all CAs. For the aircraft, it is sufficient to store the identities and corresponding public keys (ID_{CAi}, Q_{CAi}) of CAs belonging to the ATSUs, that an aircraft encounters during its journey to the destination. The corresponding private key d_a is stored in tamper-proof memory. In addition, the public key certificate $Cert$, as mentioned in the registration phase, should be requested before takeoff.

4.4. Mutual authentication and key exchange phase

We assume authentication and key exchange between the ATSU and aircraft, with identities ID_i and ID_a, respectively. Each time a new aircraft enters the zone of the ATSU, the ATSU sends its identification information to the aircraft containing $M_1 = \{ID_i, Cert_i, LT_i, ID_{CAi}, R_i, s\}$. Note that $R_i = r_iG$ is a random point on the curve, whereas s is the corresponding signature, $s = d_i - H(ID_i, Cert_i, LT_i, ID_{CAi}, T_i)r_i$ generated through the Schnorr algorithm. Additionally, unique random secrets (r_i and r_a) are used for each message exchange to serve as nonces [30,31].

The aircraft receiving message $M_1 = \{ID_i, Cert_i, LT_i, ID_{CAi}, R_i, s\}$ first verifies the freshness of the message by evaluating the random value R_i. The freshness of a random value is evaluated to ensure that each message is unique and to prevent replay attacks [32]. In other words, aircraft or ATSU examines the uniqueness of the received messages, and if not unique, they terminate the session.

If fresh, it continues with the process of checking the validity of the signature. Afterward, it derives the public key of the ATSU by $H(ID_i, Cert_i, LT_i)Cert_i + Q_{CAi}$. Then, it checks the equality in Eq. (1):

$$sG \equiv Q_i - H(ID_i, Cert_i, LT_i, ID_{CAi})R_i \quad (1)$$

If they are equivalent, the aircraft continues the process. Otherwise, it terminates. Now, there are two options.

(1) Case 1: When $ID_{CAi} \equiv ID_{CAa}$, the aircraft knows that the ATSU is aware of the public key belonging to ID_a. It suffices for the aircraft to send $M_{2a} = \{R_a, E_K(ID_a), H(K^*), s_a\}$. Here $R_a = r_aG$ is a random point on the curve used to derive the session key $K = r_aR_i$. Further, the signature s_a is defined by $s_a = r_a - H(K, ID_a)d_a$. Note that K^* denotes the secret session key derived with the previous ATSU, which equals d_aR_i in case of take-off.

(2) Case 2: If $ID_{CAi} \neq ID_{CAa}$, the aircraft must also communicate certificate-related material. As a consequence, the aircraft sends the message $M_{2b} = \{R_a, E_K(ID_a, Cert_a, LT_a), ID_{CAa}, H(K^*), s_a\}$. Here $R_a = r_aG$, $K = r_aR_i$ and $s_a = r_a - H(K, ID_a, Cert_a, LT_a, ID_{CAa})d_a$.

Upon the arrival of the message M_{2a} or M_{2b}, the ATSU first verifies the freshness of the message by evaluating the random value R_a. If fresh, the ATSU derives the Diffie–Hellman secret session key $K = r_iR_a$ and decrypts the message. The data $H(K^*)$ is used to verify the successful handover phase. As illustrated in Fig. 3, it has either been sent by the previous ATSU, or else it equals r_iQ_a in case of take-off of the flight. In both options, the ATSU can retrieve the public key Q_a belonging to ID_a, i.e., by looking it up in the local database or by using the ECQV derivation formula, respectively. Next, the signature is verified by checking the equality $s_aG \equiv R_a - H(K, ID_a, Cert_a, LT_a, ID_{CAa})Q_a$. Equivalency proves the aircraft's legitimacy and lends credibility to the freshly created session key K. The initial value of K^* is precalculated and stored in the ATSU for each flight departing its control area. Note: In case the signature is correct, but the $H(K^*)$ is incorrect, the flight plan needs to be verified, and an error message containing a request to send the previous ATSU ID must be sent to the aircraft.

4.5. Handover phase

In case $ID_{CAi} = ID_{CAa}$, the ATSU is aware of the aircraft's flight pattern and thus can send a handover request in advance to the next ATSU. After generating a shared session key K, the receiving ATSU verifies the truthfulness of the $H(K^*)$ by comparing the values obtained from the previous ATSU and the aircraft. Since only the legitimate aircraft and previous ATSU knows the session key K^*, the next ATSU is guaranteed a safe handover.

In most cases, when $ID_{CAi} \neq ID_{CAa}$, for instance, after crossing the ocean, the related EU and USA ATSUs typically do not send a handover request. However, due to the certificate format, the new ATSU can still verify the validity of the aircraft by relying on the information provided by the corresponding CA.

5. Security analysis

In this section, we did a formal and informal analysis of AKAASH, discussed in the following section in more detail.

```
Verification summary:

Query not attacker(Ki[]) is true.

Query not attacker(ri[]) is true.

Query not attacker(IDa[]) is true.

Query not attacker(ra[]) is true.
```

Fig. 5. ProVerif simulation results.

5.1. Formal analysis using proverif

We implemented the proposed scheme on Proverif to check its robustness against cyber attacks. Proverif is a widely utilized tool for conducting formal security assessments, utilizing the well-known Dolev-Yao attack model in its evaluations. Proverif is equipped to handle various cryptographic primitives, including shared and public-key cryptography (such as encryption and digital signatures), hash functions, and Diffie–Hellman key agreements. It performs secrecy and authentication checks on the protocol under evaluation. We consider two communication channels in AKAASH: the private channel $ChSec$ and the public channel $ChPub$. The $ChSec$ is used to communicate between an ATSU and an aircraft during the registration phase, whereas the $Chpub$ is used for communication during the authentication phase.

All participants in the Proverif system employ cryptographic methods to compute and validate session keys. The cryptographic functions, such as hash, concatenation, encryption, and decryption, are defined using Proverif's built-in constructs. The results of the Proverif analysis are displayed in Fig. 5. The evaluation of the following queries is highlighted in summary: (i) The query, not attacker(K_i[]), is true and confirms that the key "K_i[]" has not fallen into the hands of any malicious actors, and its confidentiality remains intact. (ii) The query, not attacker(r_i[]) and (r_a[]), is true, verifies that the value "r_i[]" and "r_a[]" have not been compromised, and the attacker cannot guess/brute force. (iii) The query, not attacker(ID_a[]), is true, demonstrating that the identifier "ID_a[]" has not been exploited by any malicious entities, and its security remains uncompromised.

It can be concluded that all of our protocol's events have started and terminated successfully. Using ProVerif, we demonstrated that AKAASH security approach assures secure authentication, key exchange, and handover. For readers interested in further exploring and understanding our findings, the source code of Proverif has been made available in [33] for use and reproduction.

5.2. Informal analysis

For the security analysis, we mainly focus on the authentication and key establishment phase, as the registration and initialization rely on well-known principles like the ECQV protocol and the existence of tamper-proof memory. Furthermore, we assume that the ATSUs possess a secure channel to communicate over, which can be established by mechanisms like, for instance, SSL. In the scheme, we assume the existence of an eavesdropping attacker during the authentication and key agreement phase who may also be active and able to manipulate the messages sent over the channel. We now briefly discuss how the main security features are being established.

Theorem 1. *AKAASH ensures Mutual Authentication.*

Proof. AKAASH allows only legitimate parties, such as the ATSU and aircraft, to participate in the communication. AKAASH uses ECQV-based lightweight implicit certificates to verify the authenticity of the entities involved in the communication. For example, if the aircraft receives the message $M_1 = \{ID_i, Cert_i, LT_i, ID_{CAi}, R_i, s\}$ from the ATSU, it can examine the authenticity of the ATSU by computing $sG \equiv Q_i - H(ID_i, Cert_i, LT_i, ID_{CAi})R_i$. If the expressions on either side do not match, authentication fails. Similarly, the ATSU can verify the legitimacy of the aircraft by extracting information from $M_{2a} = \{R_a, E_K(ID_a), H(K^*), s_a\}$ and computing $s_aG \equiv R_a - H(K, ID_a, Cert_a, LT_a, ID_{CAa})Q_a$. Because sensitive information such as ID_a, $Cert_a$, and K is not shared as plain text, no one, including an attacker, can impersonate a legitimate party and perform authentication. □

Theorem 2. *AKAASH preserves the property of Integrity.*

Proof. It is practically impossible to modify the messages in AKAASH due to the use of hash functions. Consider a scenario in which the attacker tries to modify the lifetime LT_i of the expired certificate for misuse, $M_{1A} = \{ID_i, Cert_i, LT_{iA}, ID_{CAi}, R_i, s\}$. Upon receiving, the aircraft verifies the integrity of the message M_1 by computing $sG \equiv Q_i - H(ID_i, Cert_i, LT_i, ID_{CAi})R_i$. Since M_1 contains the message digest of LT_i in s ($= d_i - H(ID_i, Cert_i, LT_i, ID_{CAi})r_i$), the aircraft detects the alterations and terminates the session. Thus preventing modification attacks on the AKAASH and CPDLC. Similarly, the other messages M_{2a} and M_{2b} preserve the integrity and are resistant to modification attacks. □

Theorem 3. *AKAASH assures confidentiality.*

Proof. AKAASH ensures the confidentiality of sensitive information, such as the aircraft's identity (ID_a) and session keys (K). Let us assume that the attacker tries to capture the aircraft's response message $M_{2b} = \{R_a, E_K(ID_a, Cert_a, LT_a), ID_{CAa}, H(K^*), s_a\}$. In M_{2b}, the identity-related information (ID_a) is encrypted ($E_K(ID_a, Cert_a, LT_a)$), and the secret session key (K) is hashed ($H(K^*)$) before being sent. Due to the collision-resistant property of hash functions and the non-availability of the secret key (K), the attacker would not be able to retrieve any sensitive information from the message M_{2b}. Likewise, M_{2a} also preserves the confidentiality of sensitive information. □

Theorem 4. *AKAASH promises anonymity.*

Proof. It is essential to conceal the identity of the aircraft in order to ensure the communication's anonymity and fend off cyberattacks like traffic analysis and impersonation. To achieve anonymity, the AKAASH scheme never asks the aircraft to share their identity details in plain text. Let us assume a case wherein the attacker captures the message $M_{2a} = \{R_a, E_K(ID_a), H(K^*), s_a\}$ and tries to recover the identity details (ID_a). Since the identity details are encrypted ($E_K(ID_a)$) using the secret key (K), which is only known to aircraft and ATSU, there is no chance of the attacker determining the correct identity (ID_a). Therefore, anonymity is preserved. The same level of anonymity is guaranteed in message M_{2b}. □

Theorem 5. *AKAASH guarantees perfect forward and backward secrecy.*

Proof. It is assumed that an adversary is attempting to eavesdrop on the message $M_{2a} = \{R_a, E_K(ID_a), H(K^*), s_a\}$. However, it is implausible that the adversary can retrieve the secret key $K(= r_aR_i)$, as it is derived using a random secret (r_a), and only the aircraft possesses it. Moreover, the session key is hashed $H(K^*)$ before being sent. Therefore, it is challenging for an attacker to recover the key. Even if the adversary somehow obtains the secret key (K), it will only compromise the current session and not any previous or future sessions. This is because the keys used in each session are unique and have no connection to each other. For each session, unique random secrets (r_i and r_a) are used, and there is no correlation between prior and subsequent secrets, ensuring that the adversary cannot predict future or past secret keys. In conclusion, the system's security is ensured through perfect forward and backward secrecy, even if the present key is compromised. □

Theorem 6. *AKAASH is resistant to MITM attack.*

Proof. Suppose that an attacker has intercepted the message $M_{2b} = \{R_a, E_K(ID_a, Cert_a, LT_a), ID_{CAa}, H(K^*), s_a\}$. The attacker's objective is to extract confidential or personally identifiable information from the message to impersonate a legitimate entity and conduct a MITM attack. However, the attacker will be unsuccessful in retrieving any precious information, because the key to deciphering the information ($ID_a, Cert_a, LT_a$) is never shared as plain text, H(K^*).

In case the attacker wildly tries to modify the content of M_{2b}, such as R_a, $H(K^*)$, ATSU can easily detect it by computing the $H(K^*)_i \equiv H(K^*)_a$ and $s_aG \equiv R_a - H(K, ID_a, Cert_a, LT_a, ID_{CAa})Q_a$. Consequently, ATSU terminates the session, preventing the attacker from carrying out a MITM attack. It is worth noting that other messages, such as M_1 and M_{2a}, are also immune to MITM attacks. □

Theorem 7. *AKAASH is secure from impersonation attacks.*

Proof. In the event that an attacker intercepts the message $M_{2b} = \{R_a, E_K(ID_a, Cert_a, LT_a), ID_{CAa}, H(K^*), s_a\}$, there is a risk that sensitive information could be obtained, which would allow the attacker to impersonate a legitimate aircraft. However, the session key $H(K^*)$ is hashed, and certificate-related details, including identity, are encrypted, $E_K(ID_a, Cert_a, LT_a)$. The attacker cannot retrieve this information, and they are also unable to modify the value of $s_a = r_a - H(K, ID_a, Cert_a, LT_a, ID_{CAa})d_a$. As a result, the attacker cannot prove their legitimacy without access to personally identifiable information such as the ID_a or K. These security measures make the proposed AKAASH approach resistant to impersonation attacks. □

Theorem 8. *AKAASH is secure against Known key attack.*

Proof. Consider that an attacker has intercepted previous message exchanges and is trying to retrieve information related to secret keys (K) to create duplicate ones. However, as stated before, the secret keys in AKAASH use a unique and independent random secret, such as r_i and r_a, for each certificate, making the future secret key(s) $K_f = r_{af}R_{if}$ different and independent. Even if an attacker somehow obtained an old secret key (K), they would not be able to construct a new secret key (K_f) as the knowledge of r_{af} and r_{if} is not available to them. This means that knowing past secret keys does not allow the attacker to initiate new sessions. Thereby protecting the protocol against known key and ephemeral secret leakage attacks. □

Theorem 9. *AKAASH is resistant to replay attacks.*

Proof. The proposed approach is resistant to replay attacks. Let us take a scenario where the intruder captures the message $M_{2a} = \{R_a, E_K(ID_a), H(K^*), s_a\}$. Here $R_a = r_aG$ is a random value that is generated using a random secret (r_a). Suppose an intruder attempts to impersonate a legitimate aircraft by relaying a message (M_{2a}) to the ATSU. However, since the message includes an old random secret (r_a), the ATSU halts the process and immediately terminates the session. The other messages M_1 and M_{2b} are also protected from replay attacks. □

Table 3
Security feature comparison of AKAASH protocol vs. conventional protocols.

Security features	[6]	[15]	[17]	[34]	AKAASH
Authentication	✓	✓	✓	✓	✓
Integrity	✓	✓	✓	✓	✓
Confidentiality	×	×	✓	×	✓
Anonymity	×	×	×	×	✓
Forward secrecy	×	×	×	×	✓
Injection	✓	×	✓	×	✓
Replay attack	×	✓	✓	✓	✓
MITM	✓	×	✓	×	✓
Masquerading attacks	✓	✓	✓	×	✓
Eavesdropping	×	×	✓	×	✓

✓: Secure against attack/Preserve a security attribute, ×: Vulnerable/non accomplishment of the security attribute.

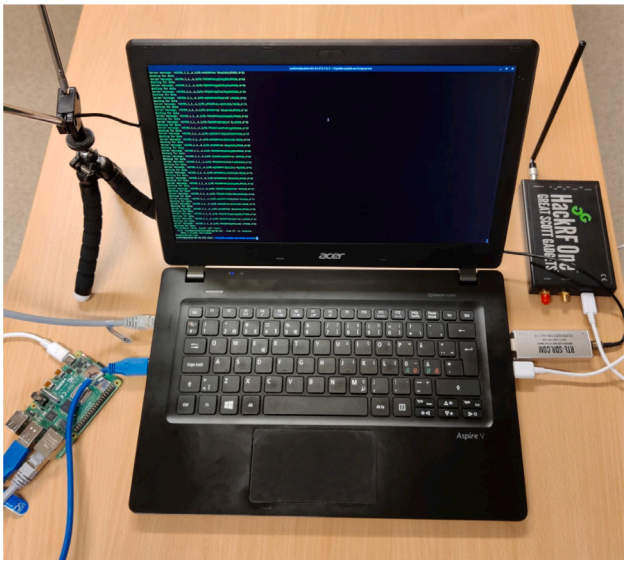

Fig. 6. Experimental setup.

In addition, we compared the AKAASH security services with the other related schemes, such as [6,15,17,34]. Table 3 shows that the proposed scheme can resist a variety of security threats to ground-air communication. On the other hand, the protocols provided in [6,15,17, 34] are unable to ensure some of the critical security features.

6. Performance evaluation

The AKAASH approach is assessed based on several parameters, including computational and storage efficiency, communication expenses, and security features. The results of these evaluations are presented in this section as follows:

6.1. Experimental setup

Fig. 6 illustrates the experimental setup for AKAASH. For the simulation of AKAASH, two different hardware platforms were used: (i) a laptop serving as the ATSU and (ii) a Raspberry Pi serving as the aircraft.

For the performance evaluation of AKAASH, we selected the open-source BouncyCastle (BC) library version 1.60 [35]. The BC library offers lightweight cryptographic primitives, including certificate generation, that are based on widely accepted standards. Furthermore, the Edwards curve is utilized for ECC operations. The BC library is simulated on two systems: a Raspberry Pi (R-PI) 3B+ with a 1.4 GHz processor and a Personal Computer (PC) with an Intel Core i7-8750H CPU running at 2.2 GHz and 16 GB of RAM.

Table 4
Execution time (in μs) of AKAASH protocol on different hardwares.

Device	R-PI	PC
Library	BC	BC
A	156.8 ±0.7	4.866 ±0.005
M	32800 ±20	1148.2 ±0.8
H	13.52 ±0.04	1.023 ±0.002
E / D	53.86 ±0.17	2.90 ±0.03

BC: Bouncy Castle, A: EC Addition, M: EC Multiplication, H: Hash (SHA-256), E/D: Encryption/Decryption(AES), R-PI: Raspberry-Pi (aircraft), PC: Personal Computer (ATSU).

6.2. Computational and storage cost

As registration occurs only once, its computational and communication costs have been disregarded. Our focus is solely on the computational and communication expenses incurred during the authentication and key exchange phase. Table 4 represents the average time for most of the computing operations of AKAASH, i.e., EC Addition (A), EC Multiplication (M), SHA-256 function (H), and Advanced Encryption Standard (AES), on different hardware platforms.

The Raspberry Pi 3B took 156.8 μs to perform addition, 32800 μs for multiplication, 13.52 μs for hashing, and 53.86 μs for encryption and decryption. In comparison, PC took 4.86 μs for an addition operation, 1148.2 μs for a multiplication operation, 1.02 μs for hashing, and 2.90 μs for encryption and decryption, respectively. The results indicate that the time required to implement AKAASH on these devices is reasonable.

It is clear from the comparison in Table 5 that the proposed AKAASH protocol is more efficient in terms of computational cost when compared to the scheme proposed in [17]. The authentication and key agreement phase in AKAASH execute fewer operations than the scheme in [17]. Additionally, AKAASH has a lower computational time for both Raspberry PI and PC when compared to the scheme in [17]. It is worth noting that the schemes proposed in [6,15,34] have not provided any computational costs for their cryptography operations, which makes it difficult to compare their performance with our proposed solution. In conclusion, the proposed solution in this paper provides a more efficient and secure solution for authentication, session key agreement, and handover compared to the existing literature, especially when compared to the scheme in [17].

Storage Cost: AKAASH requires 1056 bits of storage space for various cryptographic operations, including 32 bits for identification, 256 bits for a random number, 256 bits for hashing, 256 bits for ECC operations, and 256 bits for certificates. AKAASH ensures multiple layers of security to protect sensitive information. Authors in [17] also proposed a scheme for mutual authentication and a secure handover that requires equivalent storage space. However, it lacks some of the security goals listed in Table 3. Therefore, the AKAASH approach is more robust compared to [17].

Table 5
Computation cost comparison of AKAASH vs. conventional protocols.

A	Entities	Operations	R-PI (ms)	PC (ms)
[6]	ATSU	–	–	–
	AC	–	–	–
[15]	ATSU	–	–	–
	AC	–	–	–
[17]	ATSU1	3H + 4M + 1A + 1E + 1D	–	4.60
	AC	4H + 6M + 1A + 2E + 1D	203.08	–
	ATSU2	3H + 6M + 1A + 1D	–	6.90
	TC	10H + 16M + 3A + 3E + 3D	203.08	11.50
[34]	ATSU	–	–	–
	AC	–	–	–
AKAASH	ATSU	2H + 5M + 2A + 1D	–	5.75
	AC	4H + 6M + 3A + 1E	197.37	–
	TC	6H + 11M + 5A + 1D + 1E	197.37	5.75

A: Approach, H: Hash, M: EC Multiplication, A: EC Addition, E: Encryption, D: Decryption, Dash(-): Not Disclosed, TC: Total Cost, PC: Personal Computer (ATSU), R-PI: Raspberry-Pi (aircraft).

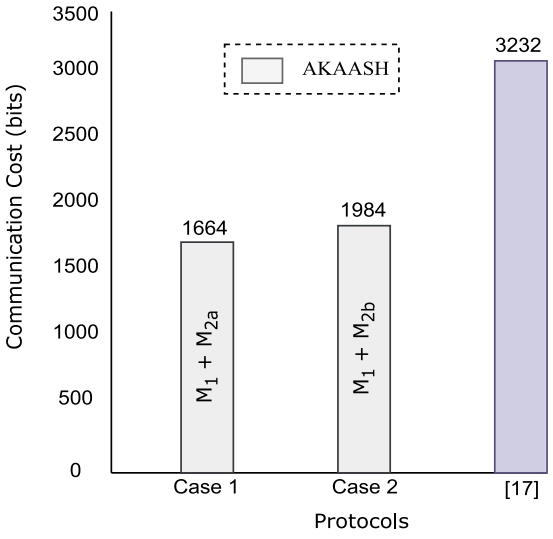

Fig. 7. Communication cost comparison of AKAASH protocol vs. conventional protocol.

6.3. Communication cost

The proposed AKAASH protocol is evaluated for its communication overhead by considering the total number of bits transmitted and received between the ATSU and the aircraft. The ATSU sends message $M_1 = \{ID_i, Cert_i, LT_i, ID_{CAi}, R_i, s\}$, which is 864 bits in size, to the aircraft. The aircraft then responds with either message $M_{2a} = \{R_a, E_K(ID_a), H(K^*), s_a\}$, which is 800 bits, or message $M_{2b} = \{R_a, E_K(ID_a, Cert_a, LT_a), ID_{CAa}, H(K^*), s_a\}$, size of 1120 bits. The total communication cost of the proposed AKAASH protocol is 1664 bits when $ID_{CAi} \equiv ID_{CAa}$, and 1984 bits if $ID_{CAi} \neq ID_{CAa}$. The communication overhead of the AKAASH protocol is lower compared to the scheme proposed in [17], which requires 3232 bits for the message authentication and handover phase. It is noteworthy that the protocols presented in [6,15], and [34] have not discussed their communication overhead, making it challenging to assess their performance relative to the proposed AKAASH protocol. The comparison between the proposed AKAASH protocol and others are presented in Fig. 7. From Fig. 7, it can be seen that AKAASH has a lower communication overhead as compared to the [17]. This demonstrates that AKAASH is more efficient in terms of communication overhead as compared to the existing protocol, which is a critical consideration in real-world applications where bandwidth is limited. Additionally, reducing communication overhead results in improved time and resource utilization performance.

6.4. Integration of AKAASH within CPDLC

Integrating AKAASH with the CPDLC system demands careful attention to both message size and time efficiency for ensuring the effectiveness and safety of air traffic control. The size of messages is critical in determining the speed and accuracy of data transmission between air traffic control and aircraft. A larger message size can lead to slower transmission times and a higher likelihood of errors, potentially causing confusion or miscommunication. Time efficiency is equally important, as prompt response times are necessary for air traffic controllers to effectively manage changing conditions and avoid safety risks or disruptions in air traffic flow. To ensure optimal air traffic control performance and safety, both message size and time efficiency must be considered and optimized during the integration of AKAASH with the CPDLC system.

The message size for CPDLC can vary depending on the system implementation, such as ATN or FANS. The ICAO sets the maximum message size for CPDLC, including spaces and punctuation, as 240 characters (1920 bits) [36]. This limit ensures that messages can be transmitted quickly and effectively while providing the necessary data for air traffic control operations. As shown in Table 6, the proposed AKAASH protocol requires a maximum of 1352 bits in the payload. These 1352 bits include 1120 bits of the message (M_{2b}) and 232 bits of the VDL Mode 2 header. Consequently, the adoption of AKAASH in the CPDLC system is a practical and suitable solution due to its efficient message size requirements, which are in accordance with ICAO specifications. [36].

The AKAASH protocol involves the exchange of two messages between the ATSU and the aircraft. The first message, from the ATSU, is 864 bits, and the second message from the aircraft can be either 800 or 1120 bits. In this case, we consider the worst-case scenario where the aircraft message is 1120 bits long. The data transfer rate for VDL Mode 2 communication is 31500 bits per second (31.5 Kbps) [37]. More details about VDL Mode 2 structure can be found in [38]. The time it takes to transmit a message is calculated as (Message Size + Header Size) / Data Transfer Rate. It takes approximately 0.034 s to transmit the ATSU's message, 0.031 s for the aircraft's 800-bit message, and 0.043 s for the aircraft's 1120-bit message shown in Table 6. In the worst-case scenario, the total communication time between the ATSU and the aircraft would be 0.07762 s. The AKAASH protocol plays a crucial role in air traffic management systems by efficiently and securely exchanging messages between the ATSU and the aircraft. The VDL Mode 2 encoding helps ensure messages' accuracy and quick transmission, thereby improving the overall performance of CPDLC systems.

In conclusion, the AKAASH protocol is a suitable solution for the CPDLC system as it meets the message size requirements set by the ICAO with a maximum payload size of 1352 bits. The efficient communication between the ATSU and the aircraft, with a maximum

Table 6
CPDLC message size (in bits) after AKAASH integration.

M	MS	RS FEC	AVLC frame	8208(x.25) header	TS	ML	TR (s)
M_1	864					1096	0.035
M_{2a}	800	16	104	24	88	1032	0.033
M_{2b}	1120					1352	0.043

M: Message, MS: Message Size, RS FEC: Reed Solomon Forward Error Correction, AVLC: Aviation VHF Link Control, TS: Training Sequence, ML: Message Length, TR: Transmission Rate.

transmission time of 0.07762 s, is facilitated by the AKAASH protocol and the VDL Mode 2 encoding. This helps to ensure the accuracy and quick transmission of messages, thereby improving the overall performance of CPDLC systems.

7. Conclusion and future work

CPDLC is now widely used as a digital communication mode for ATC due to its superior resilience and bandwidth efficiency compared to traditional voice-based communication. However, CPDLC encounters security challenges such as eavesdropping, spoofing, man-in-the-middle attacks, message replay, and impersonation attacks, which can harm individuals, service providers, and the aviation industry. Therefore, there is a need for robust security solutions. To address these security concerns, we propose a lightweight and robust solution called AKAASH that leverages AES, ECQV, and message digest. Proposed AKAASH provides mutual authentication, key establishment, and handover mechanisms to secure CPDLC communications. Additionally, we incorporate an off-flight certification feature to enhance reliability and operational efficiency.

AKAASH incurs a computational cost of 5.75 ms and 197.37 ms on the ATSU and aircraft sides, respectively. Communication from ATSU to the aircraft takes a maximum message length of 1096 bits, transmitted in just 0.035 s. In contrast, the maximum message length from the aircraft to ATSU is 1352 bits, transmitted in 0.043 s. AKAASH has undergone rigorous formal and informal security analyses and tests on reliable hardware platforms, demonstrating its exceptional effectiveness and robustness against various security threats. AKAASH can easily integrate into the CPDLC framework, making it a practical and ideal choice for securing CPDLC-enabled aviation communication networks. In the future, we aim to detect message injection and modification by identifying abnormal patterns in CPDLC messages using deep learning algorithms like auto-encoders.

CRediT authorship contribution statement

Suleman Khan: Conceptualization, Methodology, Software, Validation, Formal analysis, Implementation, Writing – original draft, Writing – review & editing. **Gurjot Singh Gaba:** Conceptualization, Methodology, Software, Validation, Formal analysis, Writing – original draft, Writing – review & editing. **An Braeken:** Methodology, Formal analysis, Writing – original draft, Writing – review & editing. **Pardeep Kumar:** Methodology, Formal analysis, Writing – original draft, Writing – review & editing. **Andrei Gurtov:** Methodology, Resources, Writing – original draft, Writing – review & editing, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This work was supported by Trafikverket, Sweden and Luftfartsverket, Sweden under Automation Program II. This work was also partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP), Sweden.

References

[1] Economic impacts of COVID-19 on civil aviation, 2022, Available: https://www.icao.int/sustainability/Pages/Economic-Impacts-of-COVID-19.aspx. (Accessed 26 July 2022).
[2] Q. Shao, M. Jia, C. Xu, X. Feng, A support system for civil aviation navigation equipment security management, Saf. Sci. 123 (2020) 104578.
[3] Evolution of technology in aviation industry, 2022, Available: https://www.spartan.edu/news/evolution-of-technology-in-aviation-industry/. (Accessed 26 July 2022).
[4] M.R. Manesh, N. Kaabouch, Analysis of vulnerabilities, attacks, countermeasures and overall risk of the Automatic Dependent Surveillance-Broadcast ADS-B system, Int. J. Crit. Infrastruct. Prot. 19 (2017) 16–31.
[5] Failure to communicate, 2022, Available: https://flightsafety.org/asw-article/failure-to-communicate/. (Accessed 26 July 2022).
[6] J. Smailes, D. Moser, M. Smith, M. Strohmeier, V. Lenders, I. Martinovic, You talkin'to me? Exploring practical attacks on controller pilot data link communications, in: Proceedings of the 7th ACM on Cyber-Physical System Security Workshop, 2021, pp. 53–64.
[7] M. Wernberg, Security and privacy of controller pilot data link communication, 2018.
[8] C.J. Roberts, ATC data link news, 2020, Available at https://members.optusnet.com.au/~cjr/introduction.htm.
[9] P. McFarlane, Developing a systems failure model for aviation security, Saf. Sci. 124 (2020) 104571.
[10] N. Mäurer, T. Guggemos, T. Ewert, T. Gräupl, C. Schmitt, S. Grundner-Culemann, Security in digital aeronautical communications a comprehensive gap analysis, Int. J. Crit. Infrastruct. Prot. 38 (2022) 100549.
[11] D. McCallie, J. Butts, R. Mills, Security analysis of the ADS-B implementation in the next generation air transportation system, Int. J. Crit. Infrastruct. Prot. 4 (2) (2011) 78–87.
[12] M. Strohmeier, Security in Next Generation Air Traffic Communication Networks (Ph.D. thesis), University of Oxford, 2016.
[13] S. Eskilsson, H. Gustafsson, S. Khan, A. Gurtov, Demonstrating ADS-B and CPDLC attacks with software-defined radio, in: 2020 Integrated Communications Navigation and Surveillance Conference, ICNS, IEEE, 2020, pp. 1B2–1.
[14] A. Gurtov, T. Polishchuk, M. Wernberg, Controller–Pilot data link communication security, Sensors 18 (5) (2018) 1636.
[15] T. McParland, V. Patel, W. Hughes, Securing air-ground communications, in: 20th DASC. 20th Digital Avionics Systems Conference (Cat. No. 01CH37219), Vol. 2, IEEE, 2001, pp. 7A7–1.
[16] D. Getachew, J. Griner, An elliptic curve based authentication protocol for controller-pilot data link communications, Int. J. Comput. Sci. Netw. Secur. (2005).
[17] S. Khan, A. Gurtov, A. Breaken, P. Kumar, A security model for controller-pilot data communication link, in: 2021 Integrated Communications Navigation and Surveillance Conference, ICNS, IEEE, 2021, pp. 1–10.
[18] Hack attack grounds airplanes, 2022, Available: https://www.bankinfosecurity.com/hack-attack-grounds-airplanes-a-833/. (Accessed 26 July 2022).
[19] S. Gunawardena, J.M. Rankin, Controller-pilot communications using a VDL mode 2 datalink for the NASA runway incursion prevention system, in: 20th DASC. 20th Digital Avionics Systems Conference (Cat. No. 01CH37219), Vol. 1, IEEE, 2001, pp. 2D4–1.
[20] G.S. Gaba, M. Hedabou, P. Kumar, A. Braeken, M. Liyanage, M. Alazab, Zero knowledge proofs based authenticated key agreement protocol for sustainable healthcare, Sustainable Cities Soc. 80 (2022) 103766.
[21] H. Benaddi, M. Jouhari, K. Ibrahimi, A. Benslimane, E.M. Amhoud, Adversarial attacks against IoT networks using conditional GAN based learning, in: GLOBECOM 2022-2022 IEEE Global Communications Conference, IEEE, 2022, pp. 2788–2793.

[22] E. Ukwandu, M.A. Ben-Farah, H. Hindy, M. Bures, R. Atkinson, C. Tachtatzis, I. Andonovic, X. Bellekens, Cyber-security challenges in aviation industry: a review of current and future trends, Information 13 (3) (2022) 146.
[23] S. Khan, G.S. Gaba, A. Gurtov, A federated learning based security for controller pilot data link communication, in: 33rd Congress of the International Council of the Aeronautical Sciences, ICAS, Stockholm, Sweden, ICAS, 2022, pp. 1–13.
[24] N. Koblitz, Elliptic curve cryptosystems, Math. Comp. 48 (177) (1987) 203–209.
[25] M. Lavanya, V. Natarajan, Lightweight key agreement protocol for IoT based on IKEv2, Comput. Electr. Eng. 64 (2017) 580–594.
[26] P.S. Barreto, M.A. Simplicio, J.E. Ricardini, H.K. Patil, Schnorr-based implicit certification: Improving the security and efficiency of vehicular communications, IEEE Trans. Comput. 70 (3) (2020) 393–399.
[27] A.M. Almuhaideb, S.S. Algothami, ECQV-based lightweight revocable authentication protocol for electric vehicle charging, Big Data Cogn. Comput. 6 (4) (2022) 102.
[28] B. A., Public key versus symmetric key cryptography in client–server authentication protocols, Int. J. Inf. Secur. 21 (5) (2022) 103–114.
[29] M. Masud, G.S. Gaba, P. Kumar, A. Gurtov, A user-centric privacy-preserving authentication protocol for IoT-AmI environments, Comput. Commun. 196 (2022) 45–54.
[30] C.-S. Park, A secure and efficient ecqv implicit certificate issuance protocol for the internet of things applications, IEEE Sens. J. 17 (7) (2016) 2215–2223.
[31] A. Bruni, T. Sahl Jørgensen, T. Grønbech Petersen, C. Schürmann, Formal verification of ephemeral diffie-hellman over COSE (EDHOC), in: Security Standardisation Research: 4th International Conference, SSR 2018, Darmstadt, Germany, November 26-27, 2018, Proceedings 4, Springer, 2018, pp. 21–36.
[32] M. Campagna, D. Stebila, ECMQV_ECQV cipher suites for transport layer security (TLS), Technology (2010).
[33] S. Khan, A. Gurtov, Proverif simulation for AKAASH protocol, 2023, https://github.com/sulemankhan354/Proverif.
[34] J.H. Griner, An Elliptic Curve Based Authentication Protocol for Controller-Pilot Data Link Communications, Citeseer, 2005.
[35] S. Patonico, Study and analysis of security features for internet of things devices in a onem2m-based architecture, 2020, Available at https://researchportal.vub.be/en/studentTheses/study-and-analysis-of-security-features-for-internet-of-things-de.
[36] I.C.A. Organization, Global Operational Data Link (GOLD) Manual, 2016.
[37] C.C. Insaurralde, E. Blasch, Situation awareness decision support system for air traffic management using ontological reasoning, J. Aerosp. Inf. Syst. 19 (3) (2022) 224–245.
[38] S. Lundström, Technical Details of VDL Mode 2, Rep. TSKS03, Linköping Univ., Linköping, Sweden, 2016.