UDC 004.056.55:623.746.9:681.3.06

Lightweight Cryptography in UAV systems

Maksym Skorobahatko[1], Andrii Voitsekhovkyi[1]

[1] *National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine*

Abstract

The development and adoption of lightweight cryptographic algorithms have become increasingly important due to the growing volume of data transmitted by unmanned aerial vehicles (UAVs) and other unmanned systems. These systems demand security solutions that are both efficient and resource-conscious. Lightweight cryptography offers a promising balance of performance, low energy consumption, and implementation simplicity. In this paper, we present the first comparative analysis specifically focused on lightweight cryptographic algorithms for UAVs, assessing their suitability for real-time control and data protection in resource-constrained environments. A key contribution of our work is a practical experiment measuring processor load, memory usage, and energy consumption of selected algorithms on drone hardware. The results provide a clear evaluation of each algorithm's effectiveness and efficiency under realistic operating conditions.

Keywords: Lightweight cryptography, Unmanned aerial vehicles, Theoretical and Applied Cybersecurity

Introduction

Unmanned aerial vehicles (UAVs) and other unmanned technologies are increasingly used across a wide range of domains, including agriculture, security, defense, logistics, and environmental monitoring. As the volume of data exchanged within these systems continues to grow, ensuring the security of transmitted information has become a critical challenge. UAVs are especially vulnerable to various cyber threats, including data interception, manipulation, and spoofing, which can lead to serious operational and safety consequences.

To address these risks, cryptographic mechanisms must be integrated into UAV communication systems to guarantee data confidentiality, integrity, and authenticity. Cryptography—defined as the science of secure information processing—encompasses a broad range of algorithms, protocols, and techniques designed to prevent unauthorized access and to ensure trust in digital communication [1, 2].

In recent years, lightweight cryptography (LWC) has emerged as a specialized field aimed at developing cryptographic algorithms that maintain strong security properties while being optimized for constrained environments. These lightweight algorithms are characterized by reduced memory usage, smaller key sizes, lower computational overhead, and faster execution times when compared to traditional ("heavyweight") cryptographic solutions [3–5]. This makes them particularly suitable for UAV platforms, which often operate with strict limitations on energy, processing power, and storage.

One of the major milestones in the advancement of LWC was the NIST Lightweight Cryptography Competition, which led to the identification of a set of robust and efficient algorithms designed for embedded and low-resource systems. Among the most prominent are the authenticated encryption algorithms Ascon, Elephant, GIFT-COFB, and Xoodyak, along with widely adopted lightweight hash functions such as Blake2 and Keccak.

While prior studies have evaluated these algorithms in general IoT contexts, there remains a lack of targeted analysis for UAV-specific applications. To the best of our knowledge, this paper presents the first comparative evaluation of these lightweight cryptographic algorithms in the context of UAVs, focusing on their suitability for securing real-time communications on resource-constrained aerial platforms.

In unmanned aerial vehicle (UAV) communications, cryptography plays a critical role in ensuring the confidentiality, integrity, authenticity, and availability of transmitted data. Given the unique constraints of UAVs (e.g., limited computational power, real-time requirements, and exposure to hostile environments), the following cryptographic features are most relevant:

- Reduced computational complexity;
- Small key sizes;
- Efficiency in terms of energy and memory.

The key contributions of this article include:

- An overview of lightweight cryptographic algorithms and their relevance to UAV environments;
- A focused study on selected LWC finalists (Ascon, Elephant, GIFT-COFB, Xoodyak);
- Practical implementation and testing of these algorithms on embedded hardware platforms representative of UAV systems;
- An experimental evaluation of each algorithm's performance in terms of memory usage, CPU load, execution time, and energy consumption;
- A discussion of the results with respect to algorithm suitability for real-world UAV deployment.

The findings of this study aim to support future efforts in the integration of lightweight cryptographic primitives into UAV systems, contributing to more secure and efficient aerial data communication.

1. Lightweight cryptography overview and application for UAVs

1.1. Lightweight cryptographic algorithms overview

Lightweight cryptographic algorithms are specially designed for devices with limited processing power, memory, and battery capacity. Unlike traditional cryptographic algorithms, such as AES (stands for Advanced Encryption Algorithm) or RSA (stands for Rivest, Shamir, Adleman), which require substantial computational and energy resources, lightweight algorithms are prioritized for simplicity, speed, and energy efficiency. They are particularly important in IoT (stands for Internet of Things), embedded systems, and UAVs, where hardware constraints limit the use of standard cryptographic primitives.

These modern lightweight ciphers typically use reduced block sizes, simplified rounds, hardware-friendly operations, and efficient key scheduling to meet the demands of constrained environments while still providing acceptable levels of security.

For the research were selected 4 lightweight algorithms: Ascon, Elephant, GIFT-COFB and Xoodyak. This selection was driven by a combination of factors, including their diverse design principles, promising security properties, and suitably for resource-constrained environments like those found in UAVs.

Authenticated Encryption with Associated Data (AEAD) is a cryptographic paradigm that ensures both the confidentiality and authenticity of data in a single, unified operation. In contrast to traditional encryption schemes that focus solely on data secrecy, AEAD algorithms simultaneously encrypt the message and generate an authentication tag to detect tampering or unauthorized modifications. This makes them particularly suitable for scenarios like UAV communication, where both secure transmission and integrity verification are critical [2, 6]. Lightweight AEAD schemes, such as those used in Ascon [7], Elephant [8], GIFT-COFB [9] and Xoodyak [10], are optimized to perform these dual functions efficiently on resource-constrained devices, providing robust protection with minimal computational overhead.

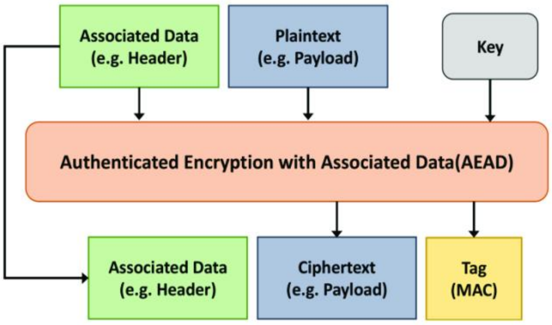

Figure 1: AEAD algorithm scheme

Ascon was chosen due to its strong performance in recent lightweight cryptography competitions, most notably its selection as a winner in the NIST Lightweight Cryptography standardization process for authenticated encryption with associated data (AEAD) and hashing. Ascon is a family of lightweight authenticated encryption and hashing algorithms, developed by Cristoph Dobrauning, Maria Eichlseder, Florian Mendel and Martin Schlaffer. It utilizes a sponge construction and permutation-based design, and provides strong resistance to differential and linear cryptanalysis.

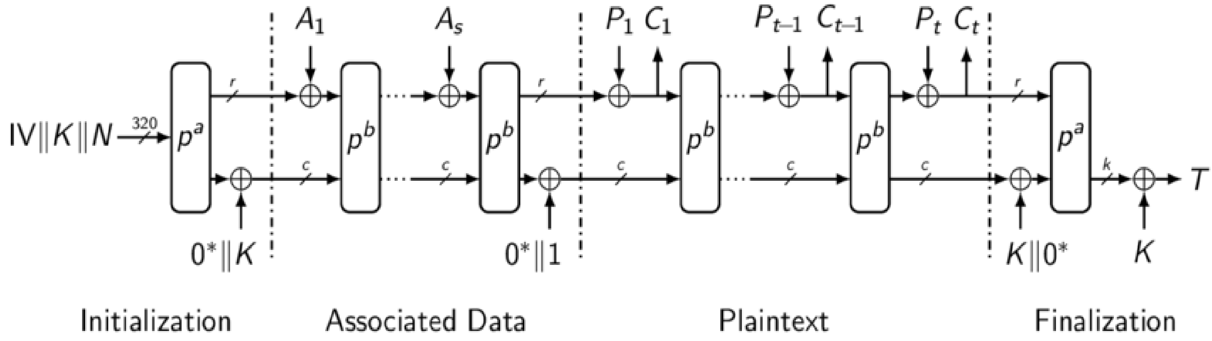

Figure 2: ASCON encryption scheme [3]

Key features include small implementation size, low latency, and ease of integration. It was also designed to be side-channel resistant [7]. The ASCON encryption scheme is presented in the Figure 2.

Elephant represents a distinct design approach, categorized as a block cipher based on a pseudorandom function (PRF). Its inclusion allowed for the exploration of a block cipher paradigm within the context of lightweight cryptography for UAV data transmission. Elephant is designed to offer a good balance between speed and security, with a focus on efficient hardware implementation. Its relative novelty also presented an opportunity to evaluate a more recent contender in the field [8].

GIFT-COFB was selected as a block-based authenticated encryption scheme known for its very small hardware footprint and energy efficiency. Its design emphasized simplicity and low power consumption, making it particularly relevant for highly resource-constrained UAV platforms. It was submitted by a team including Tetsu Iwata and Yu Sasaki, and provides an authenticated encryption scheme built on the GIFT block cipher, which itself is a successor to PRESENT, and COFB (stands for COmbined FeedBack) is a lightweight mode of operation. GIFT uses a 64-bit block size and 128-bit key size with low gate count, which makes it suitable for hardware implementation with high throughput. GIFT`s performance and security characteristics have been analyzed in various studies, providing a solid basis for comparison with other lightweight primitives [9].

Xoodyak was included as a versatile cryptographic suite based on a permutation. Developed by Joan Daemen and Gilles Van Assche (creators of Keccak), it offers both hashing and authenticated encryption functionalities within a single design. This “duplex” construction is attractive for its flexibility and potential for code reuse in systems requiring multiple cryptographic primitives. Xoodyak`s performance and security have been actively evaluated, making it a relevant and interesting algorithm to consider for securing UAV communication [10].

In conclusion, lightweight symmetric block ciphers differ from traditional ones in their fundamental design goals: minimizing gate equivalents, using compact S-boxes, reducing state size, and enabling efficient integration into hardware-constrained devices like UAVs and IoT sensors. These qualities make them especially suitable for modern embedded and real-time systems, such as drones, where performance and efficiency are critical.

2. Lightweight cryptography application in UAV data protection

Cryptography plays a critical role in ensuring the security and trustworthiness of UAV networks, especially in the UAV communications during the wartime [11]. These networks often involve communication between UAVs (drones), ground control stations (GCS), and sometimes cloud servers or peer UAVs. Given the sensitive nature of this communication and the potential consequences of compromise (e.g., hijacking, spoofing, or denial of service), cryptographic techniques are essential.

In modern warfare, unmanned aerial vehicles (UAVs) play a pivotal role in reconnaissance, target tracking, surveillance, and tactical operations. Their deployment in high-risk environments makes them a prime target for cyber and electronic attacks. To safeguard UAV missions and preserve the integrity of military operations, cryptographic techniques serve as a foundational element in defending against a wide spectrum of threats.

One of the most critical aspects that cryptography secures is the confidentiality of mission-critical data. UAVs routinely transmit sensitive information such as live video feeds, positional data, and operational commands. Without encryption, such data could be intercepted by adversaries, compromising mission secrecy and enabling the enemy to anticipate or counter planned actions. Symmetric encryption schemes, particularly those optimized for constrained environments, ensure that even if communication channels are exposed, the contents remain unintelligible to unauthorized parties.

Beyond data secrecy, the integrity of transmitted information is paramount. A small alteration to a command or telemetry message could disrupt the UAV’s operation or even redirect its mission. Cryptographic mechanisms such as message authentication codes and secure hashing functions verify that data has not been tampered with in transit. These tools ensure that commands received by the UAV are exactly those that were issued by authorized control stations, thereby thwarting man-in-the-middle attacks and spoofing attempts.

Equally important is the authentication of both the UAV and its ground control station. The battlefield is a domain where adversaries might attempt to impersonate a legitimate operator to gain control of a drone or inject false commands. Cryptography enables mutual authentication using digital signatures and certificate-based protocols, ensuring that only verified and trusted entities can initiate or modify UAV operations. This is especially crucial in contested environments where electronic warfare capabilities may be deployed to deceive or mislead automated systems.

Command and control link between a UAV and its operator is often the primary vector for attacks. By employing authenticated encryption, this communication channel can be made resistant not only to eavesdropping but also to unauthorized alterations. Secure protocols modeled after TLS or custom lightweight variants help maintain real-time integrity and confidentiality, even under the constraints of low-latency battlefield communications.

Cryptography also plays a role in countering GPS spoofing, a tactic increasingly used to misguide autonomous systems. Signed navigation messages and anti-spoofing mechanisms ensure that the UAV’s positioning and timing data are authentic and verifiable. In hostile environments, such integrity measures are essential for preserving the accuracy of navigation and coordination.

Another essential layer of protection involves safeguarding the UAV's onboard systems through secure boot processes and firmware integrity checks. These ensure that only verified, untampered firmware can be executed, thus preventing the injection of malicious code designed to subvert drone behavior. This protection extends to stored data, such as mission logs or captured images, which are encrypted at rest to ensure they remain inaccessible even if the UAV is captured by enemy forces.

In scenarios involving UAV swarms or distributed aerial systems, secure inter-UAV communication becomes necessary to coordinate movements, share sensor data, and avoid collisions. Cryptography ensures that each UAV in the swarm trusts the identity and data of its peers. Lightweight cryptographic algorithms, which are energy-efficient and hardware-friendly, are particularly well-suited for this purpose.

Moreover, cryptographic measures mitigate the risk of denial-of-service and replay attacks. The inclusion of nonces, timestamps, and session keys within communication protocols helps distinguish legitimate commands from delayed or duplicated ones, preserving the availability and reliability of UAV operations under pressure.

So, cryptography is applied for the following purposes:

- Confidentiality: Ensuring only authorized parties can access sensitive data (via encryption).
- Integrity: Detecting unauthorized modification of messages (via message authentication codes or digital signatures).
- Authentication: Verifying the identity of the communicating parties (often via digital certificates and signatures).
- Non-repudiation: Ensuring that an entity cannot deny having sent a message (via digital signatures).
- Secure Key Exchange: Ensuring keys used for encryption are exchanged securely (via Diffie–Hellman implementations (e.g., ECDH), etc.).

UAV communication typically consists of the following types of data, and each requires different cryptographic protection:

- Command and Control (C2) data need to be encrypted and digitally signed to prevent spoofing and hijacking. C2 data can include navigation commands, takeoff/landing, speed, flight path updates.
- Telemetry data, such as position, altitude, speed, battery level are often signed for integrity, but may or may not be encrypted depending on sensitivity.
- Sensor payload data (coordination in swarms, collision avoidance, shared map data) need in mutual authentication, and encryption.
- Software updates have to be digitally signed to ensure authenticity and prevent malware injection.
- Identification and authorization data, such as UAV ID, mission profile, operator credentials, should be encrypted and signed.

Summary information is provided in Table 1.

Table 1
Cryptography application and UAV communication data types

Data	Encryption / Signature	Suggested algorithm
Command & Control	Yes/Yes	Lightweight symmetric Ascon, Xoodyak
Telemetry	Optional/ Yes	ECDSA, Xoodyak
Payload data (video/images)	Yes/ Optional	Lightweight symmetric GIFT-COFB
Software updates	No/Yes	ECDSA, Ascon
Inter-UAV messages	Yes/Yes	TLS-like protocols for UAV with asymmetric lightweight cryptography, Elephant
Identity/ Credentials	Yes/Yes	Identity-based cryptography, Xoodyak

3. Experimental Setup

To emulate operational conditions, a Raspberry Pi 3 Model B+ running Debian-based Raspberry OS was integrated with UAV simulation facilitated by ArduPilot Software-In-The-Loop (SITL), an open-source platform that replicates the functionality of ArduPilot firmware without requiring physical flight hardware [12]. This platform allows full emulation of autopilot logic, sensor input, GPS positioning, and telemetry feedback, making it deal for testing algorithm performance under pseudo-realistic conditions.

MAVProxy served as the Ground Control Station, interfacing with ArduPilot SITL through the MAVLink protocol [13]. MAVProxy enabled scripting support and dynamic connection to external Python modules, which allowed seamless integration of cryptographic components for real-time data encryption during the simulation.

Additionally, Google Benchmark was employed as a standardized performance measurement framework. This open-source library provides accurate micro-benchmarking C++ code, including detailed timing statistics, multiple iterations, and customizable execution parameters. Each cryptographic implementation was compiled with Google Benchmark support, and performance was profiled using the make benchmark command. This allowed for consistent measurement of execution time and CPU efficiency, independent of external system processes [1,2].

This experimental setup further incorporated an INA219 I2C microcontroller connected between a constrained power source and the Raspberry Pi. This module was crucial for real-time measurement of voltage, current, and power consuption, providing detailed insights into the energy demands of the cryptographic operations. The INA219 was directly wired to the Li-Po Battery management module SW6106 and the Raspberry Pi via the GPIO/I2C pins, allowing for precise monitoring of the power drawn by the system during the simulations. The connection scheme was proposed.

Finally the experimental setup includes Raspberry Pi 3, 40-Pin Ribbon Cable, breadboard, INA219 sensor (used for measuring voltage, current, and power consumption in real time), Li-Po charging module (SW6106), which regulates power for Raspberry Pi and sensor circuit, USB power connection. Also connectors were used: jumper wires (male-male), Micro-USB cable, USB-to-GPIO cable.

To facilitate the performance evaluation, a Python script was developed to automate the execution of the experimental workflow. This script orchestrated the initialization of the ArduPilot SITL environment, the establishment of communication via MAVProxy, and the subsequent benchmarking of each selected lightweight cryptographic algorithm [3; 5; 7; 14-18].

The script began by launching the ArduPilot SITL instance, configuring it with a standard UAV model and a predefined initial state. Following the successful initialization of the simulated drone environment, MAVProxy was invoked to establish a Ground Control Station interface, communicating with SITL over the MAVLink protocol. This setup mirrored a typical UAV control and telemetry link, providing a realistic context for the cryptographic operations [12; 13].

The cryptographic operations within the benchmarks were configured with standard parameters for each algorithm, adhering to common practices and recommendations for their use [19-21]. It means that algorithms were benchmarked using a standard block size of 128 bits (16 bytes) and standard key length of 128 bits (16 bytes).

The Google Benchmark framework automatically handled multiple iterations of the encryption process for each algorithm, collecting detailed timing statistics and CPU utilization metrics [4; 5]. The Python script parsed the output of the benchmark executions, extracting key performance indicators such as average execution time and CPU load. This automated approach ensured a consistent and repeatable evaluation of the algorithms' performance within the simulated UAV communication context [4; 5; 7-10].

4. Experiment results

The performance evaluation of the selected lightweight cryptographic algorithms, conducted within the ArduPilot SITL simulation environment on a Raspberry Pi 3 Model B+, yielded distinct resource utilization profiles for each. The conducted research yielded several critical insights into the performance of lightweight cryptographic algorithms in constrained UAV environments.

Analysis of CPU utilization during the encryption of the test message highlighted a divergence in processing demands among the algorithms. You can see the results of the CPU load in the graph below (Figure 3).

Ascon and Elephant exhibited a notable stability and efficiency in their resource management, maintaining a relatively consistent CPU usage throughout the cryptographic operations. The initial data for the experiment are given in [22]. This suggests a predictable and potentially lower impact on other concurrent processes within the UAV system.

Conversely, GIFT-COFB and Xoodyak demonstrated more fluctuating CPU loads. This variability indicates a less uniform demand on the processing resources, which could potentially lead to performance bottlenecks or increased power consumption during peak activity periods within the UAV`s operational cycle.

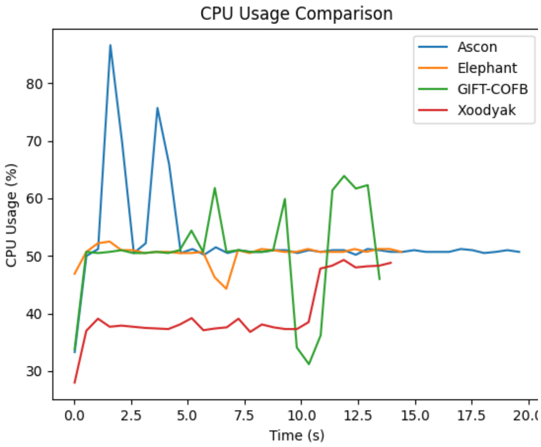

Figure 3: CPU Usage Comparison

Monitoring the memory consumption of the algorithms revealed significant differences in their RAM footprints.

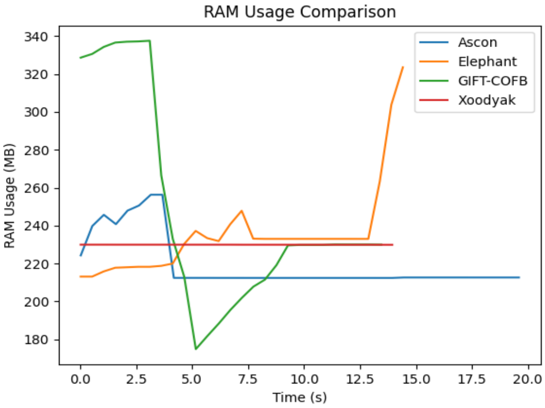

Figure 4: Memory Usage Comparison

Xoodyak emerged as the most-efficient candidate, maintaining the lowest and the most stable RAM usage throughout the testing period. This characteristic is particularly advantageous for resource-constrained embedded systems where memory availability is often limited. AScon also demonstrated relative memory usage, albeit at a slightly higher level than Xoodyak did.

In contrast, Elephant recorded the highest memory consumption among the evaluated algorithms, suggesting a potentially larger memory overhead during its operation. GIFT-COFB displayed the most volatile memory usage patterns, with significant fluctuations observed during its execution, indicating a less predictable memory demand.

Evaluation of power consumption, meticulously measured in real-time using the INA219 microcontroller, provided crucial insights into the energy efficiency of the algorithms. Ascon presented the most stable power profile, with its energy draw remaining relatively consistent over its execution time. This consistent power demand can be beneficial for predicting and managing the energy budget of a UAV`s power battery.

The other algorithms, Elephant, GIFT-COFB, and Xoodyak, exhibited significant fluctuations in their power consumption, suggesting a more dynamic but potentially less predictable energy demand. These variations could impact the overall flight time and thermal management of the UAV.

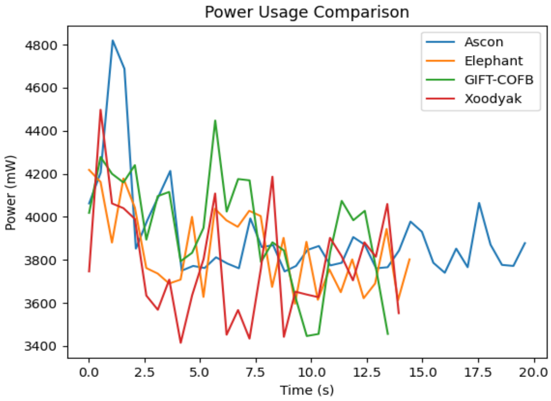

Figure 5: Power Usage Comparison

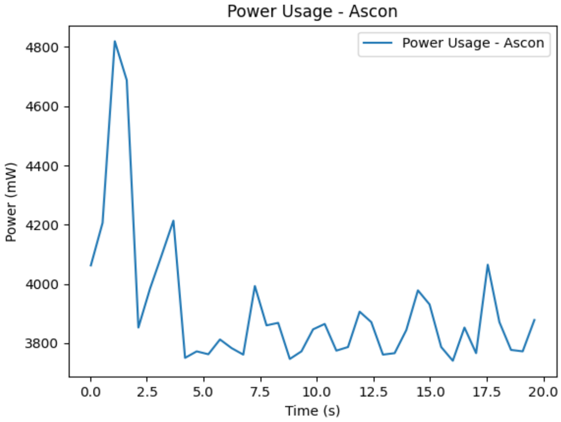

Figure 6: Ascon`s Power Usage

The analysis of the experimental runs revealed distinct operational durations for each algorithm.

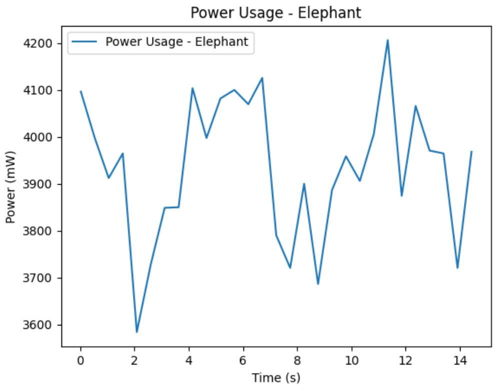

Figure 7: Elephant`s Power Usage

Ascon exhibited the longest timeframe for completing the encryption process, aligning with the benchmark data that identified it as the slowest among the evaluated candidates.

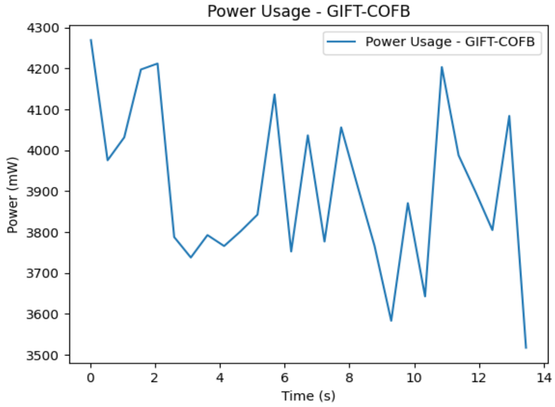

Figure 8: GIFT-COFB`s Power Usage

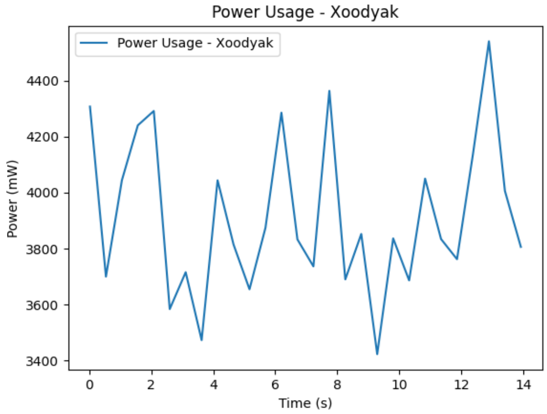

Figure 9: Xoodyak`s Power Usage

Conversely, GIFT-COFB, Elephant, and Xoodyak all demonstrated notably shorter operational durations. This observation supports the Google Benchmark findings, which indicated comparable and faster execution times for these three algorithms in comparison to Ascon. The quicker completion of the encryption task by GIFT-COFB, Elephant, and Xoodyak underscores their higher throughput within the simulated environment.

This comparison of the operational timeframes reinforces the relative speeds of the different lightweight cryptographic algorithms under evaluation, highlighting the trade-offs between execution speed and other resource utilization characteristics.

Based on the collected performance metrics - CPU load, memory usage, execution time and real-time power consumption (Fig.5-9), it is possible to draw initial conclusions regarding the potential susceptibility of the analyzed Lightweight cryptographic algorithms to timing attacks. Although these measurements do not replace formal side-channel analysis, they provide meaningful insights into the consistency and predictability of algorithm behavior under execution, which are essential factors in assessing timing resistance.

Ascon demonstrated the most stable CPU usage and power consumption across all test iterations. Its design includes considerations for side-channel resistance, and the consistency observed in both execution time and memory profile suggests that it likely operates in constant time. This makes Ascon the most favorable candidate in terms of timing attack resistance.

Xoodyak also showed low and consistent resource usage, particularly in memory and power profiles. Its duplex-based design and uniform processing pattern imply strong potential for constant-time execution, making it another reliable choice where resistance to timing-based side-channel attacks is a priority.

Elephant, while stable in terms of CPU load, exhibited noticeable memory usage fluctuations. These variations may not directly result in timing vulnerabilities, but they suggest that the internal state may not be fully independent of input parameters, which requires further inspection. However, its overall performance suggests moderate resilience to timing attacks, assuming a careful implementation.

GIFT-COFB, although the fastest among the tested algorithms, exhibited the greatest variability in both memory usage and power consumption. This behavior may be indicative of data-dependent operations, which in turn could be exploited in timing or power-based side-channel attacks. Therefore, despite its performance advantage, GIFT-COFB may require additional implementation-level hardening to ensure resistance against such threats.

Conclusions

In summary, the evaluation of lightweight cryptographic algorithms in a simulated UAV environment highlighted distinct trade-offs in their resource utilization profiles. Ascon and Elephant demonstrated consistent and stable CPU usage, while Xoodyak proved to be the most memory-efficient. Ascon also maintained a steady power consumption profile, advantageous for energy-constrained operations. Conversely, GIFT-COFB, Elephant, and Xoodyak exhibited greater variability in power demand. Regarding processing speed, GIFT-COFB achieved the highest throughput, whereas Ascon was the slowest among the evaluated algorithms. These results emphasize the importance of selecting cryptographic algorithms based on the specific performance constraints and operational requirements of the target UAV platform. The observed differences in stability, efficiency, and speed provide critical insights for the informed integration of lightweight cryptographic solutions into resource-limited aerial systems.

According to the experimental results, it is possible to assume that Ascon and Xoodyak stand out as the most robust choices with respect to timing attack resistance due to their consistent and predictable behavior. Elephant offers a balanced compromise, while GIFT-COFB, though efficient, may benefit from side-channel countermeasures to enhance its security profile in sensitive UAV networks.

Prospects for further research may include experimental investigation of the properties of other cryptographic transformations relevant for data protection of unmanned aerial vehicles.

References

[1] National Institute of Standards and Technology. (Last Updated 2024, December 15). Cryptography. (Computer Security Resource Center). [Online]: https://csrc.nist.gov/ .

[2] W. Stallings. Cryptography and Network Security. (2020). Principles and Practice (8th ed.). Pearson Education. 978-1-292-43749-1

[3] K.A. McKay, et al. (2017) Report on Lightweight Cryptography (NIST IR 8114). https://doi.org/10.6028/NIST.IR.8114

[4] A.J. Acosta, E. Tena-Sánchez, C.J. Jiménez, J.M. Mora, (2017). Power and energy issues on lightweight cryptography, J. Low Power Electron. 13, pp. 326–337. doi:10.1166/jolpe.2017.1490

[5] M. Rana, Q. Mamun, R. Islam, “Lightweight cryptography in IoT networks: A survey”, Future Generation Computer Systems. 129 (2022) pp. 77–89. https://doi.org/10.1016/j.future.2021.11.011

[6] S. Taneja, M. Alioto. (2018) Ultra-Low Power Crypto-Engine Based on Simon 32/64 for Energy- and Area-Constrained Integrated Systems. Computer Science. https://doi.org/10.48550/arXiv.1811.08507.

[7] C. Dobraunig, M. Eichlseder, F. Mendel et al. (2021). Ascon v1.2: Lightweight Authenticated Encryption and Hashing. Journal of Cryptology, 34, 33. https://doi.org/10.1007/s00145-021-09398-9.

[8] T. Beyne, Y. L. Chen, C. Dobraunig, and B. Mennink (2022). Status Update on Elephant. Nist Computer Security Resource Center. [Online]: https://csrc.nist.gov/csrc/media/Projects/lightweight-cryptography/documents/finalist-round/status-updates/elephant-update.pdf .

[9] S. Banik, S. K. Pandey, T. Peyrin et al. GIFT: A Small Present. Lectures Notes in Computer Science. International Conference on Cryptographic Hardware and Embedded Systems. doi: 10.1007/978-3-319-66787-4_16

[10] J. Daemen, G. Van Assche, S. Hoffert et al. (2020). “Xoodyak, a lightweight cryptographic scheme”. IACR Transactions on Symmetric Cryptology. doi: 10.46586/tosc.v2020.iS1.60-87

[11] O. Novikov, I. Stopochkina, A. Voitsekhovskyi, M. Ilin, M. Ovcharuk (2024). Simulation of UAV networks on the battlefield, taking into account cyber-physical influences that affect availability. Theoretical and Applied Cyber Security, Vol. 6 No. 2. P.66-76. https://doi.org/10.20535/tacs.2664-29132024.2.318182

[12] Ardupilot. [Online]: https://ardupilot.org/

[13] MAVProxy documentation. [Online]:https://ardupilot.org/mavproxy/index.html

[14] V. A. Thakor et al. (2021) Lightweight Cryptography Algorithms for Resource-Constrained IoT Devices: A Review,Comparison and Research Opportunities. IEEE Access, 9, 28178–28193. https://doi.org/10.1109/ACCESS.2021.3052867.

[15] C. Dobraunig, M. Eichlseder, M. Schläffer, F. Mendel. (2023). The Ascon family for lightweight cryptography. PBC 2023. [Online]: https://permutationbasedcrypto.org/2023/files/slides/PBC2023-Maria_Eichlseder.pdf

[16] M. Dworkin (2001). Recommendation for block cipher modes of operation: Methods and techniques (NIST Special Publication 800-38A). National Institute of Standards and Technology. [Online]: https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-38a.pdf

[17] S. Taneja, M. Alioto. (2018) Ultra-Low Power Crypto-Engine Based on Simon 32/64 for Energy- and Area-Constrained Integrated Systems. Computer Science. https://doi.org/10.48550/arXiv.1811.08507.

[18] N.M. Naser, J. R. Naif (2022). A systematic review of ultra-lightweight encryption algorithms. International Journal of Nonlinear Analysis and Applications, 13, pp. 3825–3851. https://doi.org/10.22075/ijnaa.2022.6167

[19] W. J. Buchanan, S. Li, R. Asif (2017). Lightweight cryptography methods. Journal of Cyber Security Technology, 1(3-4), 187-201. https://doi.org/10.1080/23742917.2017.1384917.

[20] R. Islam, Q. Mamun, M. Rana (2022). Lightweight cryptography in IoT networks: A survey. Future Generation Computer Systems, 129, 77-89. https://doi.org/10.1016/j.future.2021.11.011

[21] T. Chinbat et al. (2024). Machine learning cryptography methods for IoT in healthcare. BMC Medical Informatics and Decision Making, Vol. 24, Article number: 153. https://doi.org/10.1186/s12911-024-02548-6

[22] M. Skorobahatko, I. Stopochkina, (2025). Github. [Online]: https://github.com/user3719431/lightweight_cryptography_in_uav_systems/tree/main