INTERNATIONAL JOURNAL OF CIRCUITS, SYSTEMS AND SIGNAL PROCESSING
DOI: 10.46300/9106.2023.17.8 Volume 17, 2023

Secured Drone Communication Based on Esalsa20 Algorithm

[1] Ibtesam Jomaa
[1] Computer Science Department, Diyala University, 320001, Diyala, Iraq
[2] Worud Mahdi Saleh
[2] Directorate General of Education in Diyala, Ministry of Education, 320001, Diyala, Iraq
[3] Rasha Rokan Ismail
[3] Computer Science Department, Diyala University, 320001, Diyala, Iraq
[4] Saja Huzber Hussien
[4] Ministry of Higher Education and Scientific Research, Diyala University, 320001, Diyala, Iraq
Received: August 11, 2022. Revised: January 14, 2023. Accepted: February 15, 2023. Published: March 6, 2023.

Abstract—The Unmanned Aerial Vehicle (UAV) (sometimes known as a "drone") is used in a variety of fields. Unfortunately, as they become more popular and in demand, they become more vulnerable to a variety of security threats. To combat such attacks and security threats, a proper design of a robust security and authentication system based on and stream cipher lightweight salsa20 algorithm with chaotic maps is required. By using a proposed key generation method which is based on a 1d Logistic chaotic map to produce a flight session key for a drone with a flight plan, and then records the flight session key and the drone's flight plan in a central database that can be accessed. Finally, while the drone is flying, a GCS checks authentication of the current flight session based on the on flight session key and its flight plan as the message authentication code key to authenticate the drone by any flight session, and the drone after which uses salsa20 lightweight to cipher payload data to improve security Network Transfer of RTCM Messages over Internet Protocol Protocol (NTRIP) communication protocol and send it to GCS, and at last, a GCS verifies authentication of the current flight session based on the on flight session key and its flight plan as the message authentication code key to authenticate the drone. The proposed system is superior to other similar systems in terms of security and performance, according to the review.

Keywords—Unmanned Aerial Vehicle (UAV), Light-Weight Cryptography, Salsa20 Lightweight Algorithm, Network Transfer of RTCM Messages over Internet Protocol (NTRIP), 1d Logistic Function, Error Sensitivity Measurements.

I. INTRODUCTION

Drones are expected to have numerous advantages over regular vehicles, including the ability to drive at a continuous and faster speed, the absence of physical road infrastructure, route directness, and the avoidance of traffic and congestion. They are expected to shorten delivery times and improve logistics system responsiveness. These advantages of UAV-based distribution are especially noticeable in urban areas. The ensuing time and cost i) savings in commercial logistics systems could benefit both companies and customers, and ii) save lives and improve public health and safety by improving emergency services and medical supply [1]. Most autonomous drones only fly at a lower speed near hover to ensure that they can accurately sense their surroundings and have enough time to avoid obstacles. Human pilots have demonstrated that drones are capable of flying at astonishing speeds over complex terrain such as racetracks [2]. Despite the dangers and threats that manned aircraft provide to soldiers, drones were specifically created for military purposes, but they now have a wide range of additional uses. Drones are also utilized for airborne inspection and monitoring of electricity lines and oil and gas pipelines, in addition to package delivery [3]. Rather than using remote controllers, technological improvements have made it possible to control mini-drones using basic manipulations via cell phones. Drones will be used for a variety of applications, not simply commercial or personal. Law enforcement and border patrol agencies employ drones for surveillance. Drones can be used for harmful purposes as well. As a result, it is necessary to detect them and prevent them from causing damage [4]. As a result of the various attack tactics and targets, the outcomes vary. Some attacks attempt to steal data through communication connection security weaknesses, while others aim to spoof sensors, such as GPS spoofing.

In this work, we propose the security of the drone communication network for surmounting the challenging information leakage problem due to potential eavesdropping. This work aims to design an authentication system model between the drones and ground stations and make a secure channel to exchange data using lightweight algorithms. To increase the security of information transmission through drone communications with ground

control stations we use salsa20 stream cipher lightweight algorithm with two types of chaotic maps. Proposing an authentication method between the ground station and drone using proposed hash salsa20 lightweight algorithm key management.

The following are the paper's key contributors:

a) Secure the data payload of the NTRIP based on the salsa 20 algorithm.
b) Addition of an authentication method by using the proposed salsa20 lightweight key management between the grand station and drone. This new contribution will provide secure communication channels for the transmission of data between the grand station and drone.
c) The remaining part of the paper is laid out as follows: The second section delves into the related works. Section 3 discusses the work's related methods and materials, which include the proposed system's discretion Drone, key management, salsa20 lightweight algorithm, and NTRIP. The proposed design model is depicted in Section 4. The simulation recommended approach, assessment metrics, experimental outcomes, and discussion are all explained in Section 5. Lastly, in Section 6, the proposed system is brought to a conclusion.

II. RELATED WORKS

In their previous work, Jean-Paul Yaacoub et al. provided a comprehensive assessment of drone/Unmanned Aerial Vehicle (UAV) use in various domains and for varied aims [5]. A real-world assault scenario is also demonstrated, showing how the authors recreated a hacking attack on a certain drone once the hacking cycle was finished. As a result, a variety of civilian and military anti-drone/UAV countermeasures will also be investigated.

In their earlier work, Huu Phuoc Dai Nguyen et al. [6] improved cybersecurity for people and policy suggestions for future work in this field of interest. Sensors, communication lines, and photo privacy are all used to exploit cyber security holes. As a result, a mix of solutions for many sensors, through the use of secure communication links rather than Wi-Fi, and the use of the CIA trinity concepts are required to assure the security of drones.

In previous work, Abbas Yazdinejad, et al. [7] proposed a safe authentication strategy for drones in smart cities with low latency that uses block chain technology. They use a regional architecture in a drone network, and they are using a modified decentralized consensus called DDPOS (Drone-based Delegated Proof of Stake), which does not require re-authentication, for drones between zones in a smart city. The proposed architecture for the Internet of Drones aims to have a favorable impact on increased security and reduced latency (IoDs).

In the former work of Shubhani Aggarwal, et. al. [8], to lessen the burden of drones, they employed block chain technology to store acquired data from the drones and update information into distributed ledgers. It also ensures that the data collected by the drones in the proposed system is secure, authenticated, and authorized.

In his previous work, Zhihan Lv [9] compared the convolutional neural network (CNN) technique with autonomous IoD to explore the security of the Internet of Drones (IoD). Furthermore, to analyze and build a better system security performance model, wireless communication technology was used. According to the results of the IoD performance investigation, the clustering technique based on signal energy has the greatest result.

This research has mostly focused on getting the power to control drones by establishing confidence between drones and ground stations, which are regarded to be trusted authorities. Furthermore, the goal of these studies is to improve the security of the communication channel between the drone and the ground station by employing classical cryptographic algorithms that utilize Drone resources like time and energy. Unlike these approaches, the goal of our research is to determine whether a drone in a fly zone has been granted permission to fly through authentication between drones and ground stations, as well as to improve the security of the communication channel between drone and ground station using the lightweight stream cipher salsa20 algorithm.

III. RESEARCH AND MATERIALS

We introduce the techniques and methods employed in the suggested system to aid in the comprehension of the proposed system. We also showed the error sensitivity metrics that we used to evaluate the suggested system's performance in this work.

A. Drone

1. Security of Drones Communication

Flying items provide a high risk of damage; it is critical to understand the security and safety risks associated with UAVs. Attackers can "simply" take control of a UAV using ordinary "hacking" tools, prohibiting it from carrying out its duties or, even worse, especially damaging. As a result, there is a pressing need to improve UAV security. Drone manufacturers rely on frequency hopping, spectrum spreading, and key sharing as active security measures, therefore they are closely connected to protocols like IPv4 but do not employ security measures, leaving them open to known assaults [3]. Figure 1 depicts the UAV communication link [10].

Fig. 1. UAV communication link [10].

2. Secure Communication Protocol for Drones

NTRIP (Network Transfer of RTCM Messages over Internet Protocol) is becoming a more popular technique for providing RTK service over the Internet. NTRIP casters are available all over the world, and users with proper authentication can access the RTCM. [11].

NTRIP may be easily sent via the Internet utilizing several methods, including LAN, WiFi, and the local mobile operator's cell network. In today's world, where more than one station is connected, the NTRIP protocol is commonly utilized since all of the data produced by the connected stations is post-processed by an NTRIP server, allowing them to act as a network rather than a single connected base station. Figure 2 shows the NTRIP network diagram. NTRIP technology is the most acceptable option

for the highest level of accuracy and at least 95% time availability [12].

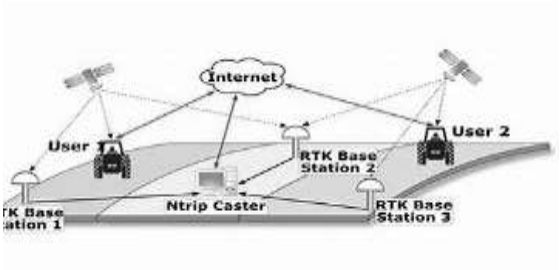


Fig. 2. NTRIP network principle diagram (courtesy of Alberding GMbH solutions) [12].

B. Lightweight Cryptography Algorithm

Light-weight cryptography is a sector of a classical cryptographic method that is commonly defined as a cryptography for resource-constrained devices. The term "light-weight encryption" is a combination of two terms "light and weight." Platforms, as well as hardware and software, use light-weight encryption and decryption. Lightweight cryptography is divided into two categories (symmetric and asymmetric cipher). Figure 3 depicts the light-weight cryptography (LWC) block diagram [13].

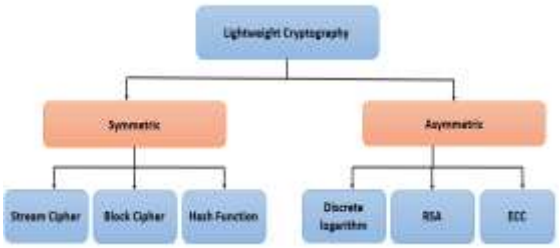


Fig. 3. Light-weight cryptography diagrams [13].

The lightweight cryptography primitives can be classified as Lightweight Block Cipher (LWBC), Lightweight Stream Ciphers (LWSC), Lightweight Hash Functions (LWHF), and Elliptic Curve Cryptography (ECC) [14]. Lightweight cryptography (LWC) expects to execute cryptographic algorithms with the use of a few computational cycles providing high robustness against security attacks meanwhile [15].

1. Salsa 20

Salsa20 is an original cipher function developed by Daniel J. Bernstein [16]. As a lightweight security technique for exchanging meter read in and other information, Salsa20 uses stream cipher as the encryption mechanism. Key stream on salsa20 methods are made up of mathematical operations that function on 32-bit words and employ a 256-bit key K = (k0, k1, k2, k3, k4, k5, k6, k7) or a 128-bit key K = (k0, k1, k2, k3) as input and a 64-bit nonce N = (n0, n1) as output, resulting in a 512-bit Keystream block sequence, see [17] for more detail.

C. 1d Logistic Function

The 1d Logistic Map, which displays simple dynamic equations numerically with complex chaotic behavior, is among the most well-known one-dimensional chaotic maps [18]. Although using 1d logistic maps increases the efficiency of cryptography algorithms [19]. The following is a description of an equation:

$$Xn + 1 = FL(u, Xn) = u \times Xn \times (1 - Xn) \quad (1)$$

Where u is the control parameter in the range u (0,4], x0 denotes the chaotic map's initial value, and Xn denotes the chaotic sequence's output [20]. The logistic map's bifurcation diagram is depicted in Figure 4. The horizontal axis in the plot represents the "r" bifurcation parameter [21].

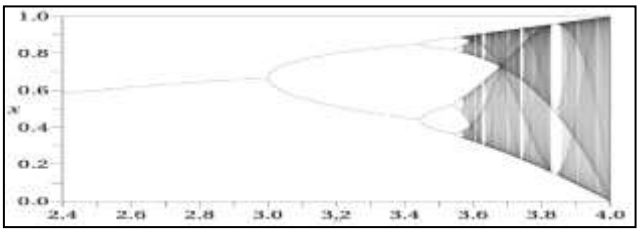


Fig. 4. The logistic map's bifurcation diagram [17].

D. Error Sensitivity Metrics

This subsection will be discussed in detail all error sensitivity measurements that using t evaluated the performance of the proposed system.

1. Mean Square Error

As indicated in equation (2), MSE is obtained by averaging the squared intensities of the original (input) and resulting (output) image pixels,

$$\mathrm{MSE} = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}((a(i,j) - b(i,j))^2}{N \times M} \quad (2)$$

The parameters a(i, j) and b(i, j) relate to the pixels in the ith row and jth column of the original and encrypted images, respectively and M×N is the image size. The lower the MSE number, the more secure the encryption [22].

2. Peak Signal-to-Noise Ratio (PSNR)

SNR is a mathematical measure of image quality depending on the pixel difference between the two images [23]. The SNR value is a comparison of the quality of the enhanced image to the original image. PSNR is given by equation (3) as follows:

$$\mathrm{PSNR} = 10\mathrm{Log}\frac{S^2}{\mathrm{MSE}} \quad (3)$$

For an 8-bit picture, s equals 255. When all pixel values are equal to the greatest possible value, the PSNR is equal to the SNR.

3. Normalization Cross Correlation

The NCC is a measurement of similarity between two wavelengths as a function of lost time applied to one of the wavelengths; the lower the value of NCC, the lower the image quality; the NCC expresses in equation(4) [24].

$$\mathrm{NCC} = \frac{\sum_{m=1}^{M}\sum_{n=1}^{N} x(m,n).y(m,n)}{\sum_{m=1}^{M}\sum_{n=1}^{N}(x(m,n))^2} \quad (4)$$

Where x is the original image, y is the encryption image and the size of images is equal to m×n.

4. Average Difference (AD)

Between the input and the recovered images, the average difference metrics of difference are calculated. If the greatest difference is high, it indicates that the image quality is weak. Equation (5) can be used to calculate AD [25].

$$AD = \frac{1}{M \times N}\sum_{i=1}^{M}\sum_{j=1}^{N}\big(x(i,j) - y(i,j)\big) \quad (5)$$

E. Correlation Coefficient Metric

The correlation coefficient is a factor that is used to determine the relationship between two variables: plaintext and encryption. This metric shows how well the suggested encryption algorithm defends against statistical attacks. As a result, cipher text must be distinct from plaintext. Equations (6, 7, and 8) are used to get the correlation coefficient [26].

$$\text{Corr Coef}(x,y) = \frac{\sum_{i=1}^{n}(x_i - \mu(x))(y_i - \mu(y))}{\sigma(x)\sigma(y)} \quad (6)$$

Where $\mu(x)$ and $\mu(y)$ are the respective means of x and y:

$$\mu(x) = \frac{1}{n}\sum_{i=1}^{n} x_i, \text{and } \mu(y) = \frac{1}{n}\sum_{i=1}^{n} y_i \quad (7)$$

The plaintext and ciphertext variables are x and y. Furthermore, the terms in the denominators (also known as the x and y standard deviations) are:

$$\sigma(x) = \sqrt{\sum_{i=1}^{N}(x_i - \mu(x))^{2,}} \text{and } \sigma(x) = \sqrt{\sum_{i=1}^{N}(y_i - \mu(y))^{2,}} \quad (8)$$

The plaintext and its encryption are identical if the correlation coefficient is one. If the correlation coefficient is equal to zero, the cipher text and plaintext are completely different (i.e. good encryption). As a result, encryption success equates to lower correlation coefficient values[27].

F. Average Security

Cipher secrecy is calculated using key equivocation (conditional entropy of key given cipher) [28]:

$$H\left(\frac{k}{c}\right) = \sum_{j=1}^{L}\sum_{i=1}^{n} q_i P_{ij} \log P_{ij} \quad (9)$$

Where qi= Pr (C=ci) and Pij = Pr (K=ki / C=ci) L and n are the key and cipher text lengths, respectively.

IV. THE PROPOSED SYSTEM

The proposed system's design goals include developing an authentication technique that creates a flight session key for such a drone and registers that key, as well as the drone's flight plan, in a centralized database for mutual authentication between both the drone and the ground control station. The registered flying session key is utilized by the ground station to authenticate the drone using a chaotic function. To improve the security of proposed authentication techniques, the salsa20 lightweight algorithm was used to cipher the registered flight session key in the centralized dataset in the ground station. Using lightweight cryptography salsa20 algorithms with NTRIP to protect data between ground stations and drones. Figure 5 depicts the basic block diagram for the proposed system. Three basic steps are included in the proposed system (Registration, encryption and decryption, and authentication stages). The proposed system is focused on a completely Ground Control Station (GCS) and a single drone with a large number of flying sessions.

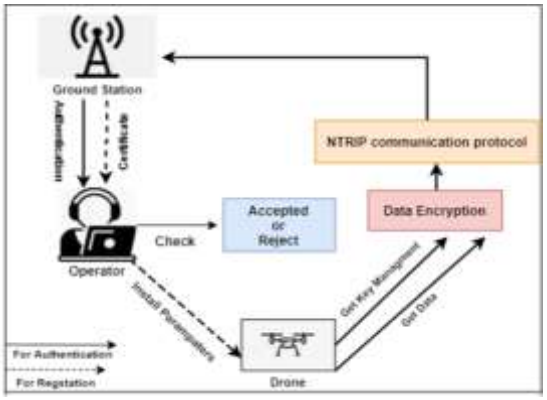


Fig. 5. The Proposed System's Primitive Block Diagram.

This is done on the GCS side when the drone intends to fly, firstly, Ground Control Station (GCS) provides a session key to the drone. The GCS gives a Certificate to the operator which is represented (computer) inside GCS, GCS allows the operator permission to begin generating the secret key for each flying session using the suggested 1d Chaotic maps depending on the key generation process. Also in this stage, the operator provides the drone with command and GPS coordinates for the current session. Figure 6: Key management technique block diagrams.

A. Registration Stage

This is done on the GCS side when the drone intends to fly, firstly, Ground Control Station (GCS) provides a session key to the drone. The GCS gives a Certificate to the operator which is represented (computer) inside GCS, GCS allows the operator permission to begin generating the secret key for each flying session using the suggested 1d Chaotic maps depending on the key generation process. Also in this stage, the operator provides the drone with command and GPS coordinates for the current session. Figure 6: Key management technique block diagrams.

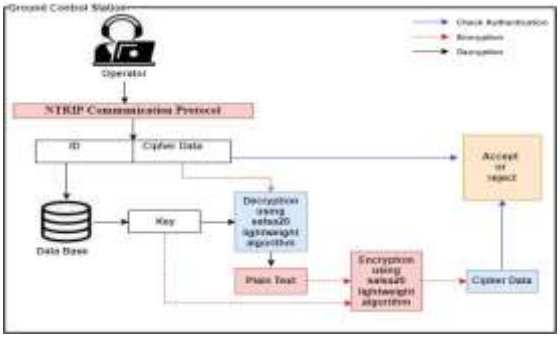


Fig. 6. The key management technique is depicted as a block diagram

1. Key management technique

When a drone initiates a flight, it sends a request to GCS for a flight session key. The GCS obtain operator, which is a computer, wishes to manage the drone's security and authentication. GCS issues a unique certificate for each drone flight session. The certificate is the permission granted by GCS to the operator to begin the key creation procedure. To create a unique and randomized flying session key without needing to do any complicated math, the proposed system depended on one dimension Logistic chaotic function as shown in detail as shown in algorithm 1.

Algorithm1. Key Generation
Input: x0, k, Key Size Output: Key

Begin
Step 1 set key[]
Index=0
Step 2 for each byte in key size do
2.1 Calculate Logistic value by equation (1)
2.2 Data =Split (Logistic value,'.') and take only digits after the dot
2.3 Convert String to integer
Look up Table = "0123456789"
Key int = 0
For each char in Data
charValue = Look upTable.Index Of (char)
keyint = (keyint * 10) + charValue
End for
2.4 Key[index]= keyint mode 255
End do
End

Better reduction algorithm is used to convert integer numbers to binary byte or (32-bits) and To Impairment Excellent for modularity reducing of huge numbers, which is based on the basic concept of avoiding the slowness of long division by involving multiplications, subtractions, and shifts, as shown in algorithm 2 [29].

Algorithm 2. Barrett modular reduction
Input: two integer number a=$(a_{2k-1}\dots a_1x_0)$b ,p=$(p_{k-1}\dots p_1p_0)$b (with Pk-1≠0),and $\mu=[\frac{b^{2k}}{p}]$ Output: r=mod p
Begin $q_1=[a/b^{k-1}]$ $q_2=q_1\mu$ $q_3=[q_2/b^{k-1}]$ $r_1=a\ mod\ b^{k+1}$ $r_2=q_3p\ mod\ b^{k+1}$ $r=r_1-r_2$ $if\ r<0\ then$ $r=r+b^{k+1}$ $while\ r\geq p\ do$ $r=r-p$ $return\ (r)$ End

This step will save the generated key in a central database in GCS as a pair of parameters for each fly. The operator requests the parameters for the drone from the database when he wants to install them. As indicated in the diagram below in Figure 7, the operator gives the drone two types of commands and coordinates:

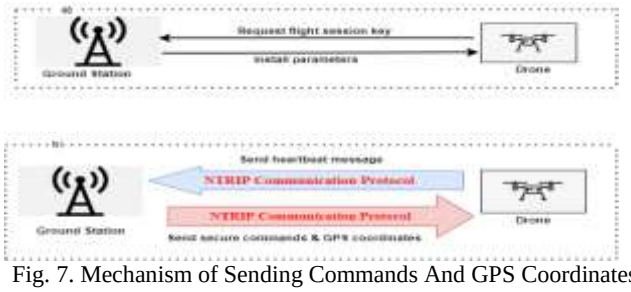


Fig. 7. Mechanism of Sending Commands And GPS Coordinates From GCS To Drone; A) Before Drone Fly, B) After Drone Fly.

The operator sets up the first command and GPS coordinates for the current flight before the drone takes to the air. After the drone fly, the operator sends a new command and GPS coordinates via NTRIP. When GCS wants to modification of the drone's path or if something malfunction occurred during the flight. A heartbeat message is transmitted from the drone to the GCS before sending any new communication through the payload to ensure that the system is ready and alive. The salsa20 lightweight method was used to encrypt data from the GCS to the drone. In addition, the checksum computed after encryption ensures that the message is received correctly by the drone. The payload is decrypted using the salsa20 stream cipher after the checksum is verified. Manually install the parameter and command in the drone.

B. Encryption Stage

This is an encryption/decryption stage that uses the salsa20 lightweight encryption algorithm to save time and energy for the drone. The suggested system was implemented in the following steps, as shown in Figure 8, to improve the security of the NTRIP communication protocol used by the Ground Control Station (GCS) and the Drone.

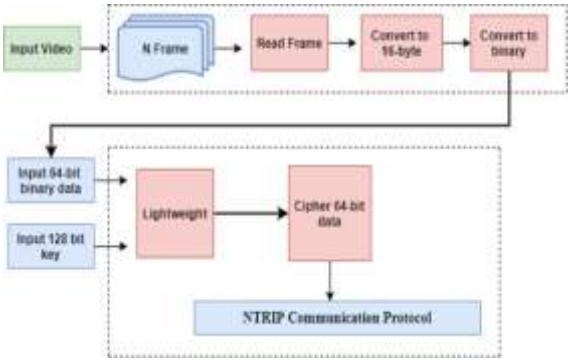


Fig. 8. Block Diagram of the Encryption Data Based On Lightweight Algorithm.

C. Authentication Stage

To develop a secure transmission channel for data exchange in the proposed system, an authentication algorithm based on the salsa20 lightweight algorithm was proposed**.** The Proposed Authentication Algorithm as explained below:

Algorithm 3. Authentication Algorithm
Input: Cipher Data Output: Authentication or Not Authentication

Begin
Step1 split Data encryption
id session = Data encryption
Data_En= Data encryption
X= Data_En
Step2 get key based on id session
Step3 decryption for each block in data
For each block in Data do
Data_dec using the salsa20 lightweight algorithm(block, key)
End for
Step4 encryption for each block in data
For each block in Data do
Data_En using the salsa20 lightweight algorithm(block, key)
End for
Y= Data_En
Step5 check authentication or not authentication
If X=y then return authentication
Else return not authentication
End

To resist different attacks, the suggested method must meet important security requirements. The following are some of the most critical requirements:

A. Mutual Authentication: For a drone and a GCS to communicate securely, the communicating entities must mutually authenticate each other.
B. Strong Key Exchange: To provide complete forward secrecy, a strong key exchange should be performed in such a way that the session keys created cannot be recovered.
C. Confidentiality: the information sent between the drone and the GCS should be kept private so that unauthorized parties cannot access it.
D. Integrity: It is important to ensure the authenticity of the information shared between the communicating ends (that the information has not been changed and that the source of the information is legitimate).
E. Non-repudiation: in such scenarios, one of the most important security needs is that the action taken by one party cannot be successfully refuted without the knowledge of others.
F. Protection against Denial of Service: Legitimate users, such as legitimate drones, should not be refused service by a service provider, such as a GCS.
G. Protection against MITM (Man-In-The-Middle) attack: the protocol prohibits an attacker from secretly relaying messages.

V. SECURITY ANALYSIS

The analysis of the proposed system security outlined in Part 4 is presented in this section. The formal security analysis is used to determine whether or not the security system meets the specified security information and requirements. Formal security analysis has been the subject of continuing for the past few years. The suggested method is explicitly proven in this study using error sensitivity-based metrics such as Mean Square Error (MSE), Peak Signal to Noise (PSNR), Normalized Cross-Correlation (NCC), and Average Differential Error (ADE) (AD). Section (5.1) illustrates the initialization of the proposed system. The implementation and results of the proposed security and authentication of drones using salsa20 lightweight encryption algorithm.

A. Initialization

The suggested system is run on a laptop computer using C#. The tests were carried out using a laptop with an Intel(R) Core(TM)i7-7700HQ processor running at 2.80GHz (8 CPUs), a 64-bit operating system, and 16384 MB of RAM. This section contains the findings of each stage of the suggested system, as well as performance evaluations based on error sensitivity measurements, correlation coefficient metrics, and average security metrics.

The suggested system consists of three main stages as illustrated in section (4) which are (Registration, encryption/decryption lightweight, and authentication). Each of these stages includes sub-steps as shown in section 5, the results of all proposed system stages clarifies in this section.

B. Results of Registration Stage

For every session of drone flying, a unique and random secret key will be generated during the registration stage. The 1d logistic chaotic function (Eq.1) is used in the suggested key generation algorithm. Table 1 shows the results of the proposed key generation algorithm in three situations, each with a different number of logistic initial parameters (x0,k) and a number of sessions (=3). The suggested key generation algorithm produces a secret key with a size of 128 bits for each drone session fly. The suggested key generation process generates a 16-byte key (16*8-128-bit) for each session, with all keys having a value between 0-255.

Table 1. Results of Key Generation Algorithm

Session Number	Key 1	Key 2	Key 3	Key 4	Key 5	Key 6	Key 7	Key 8	Key 9	Key 10	Key 11	Key 12	Key 13	Key 14	Key 15	Key 16
case #1 ($x0$=0.1 and k=1) and the number of sessions =3.																
0	109	206	9	166	49	233	114	236	111	4	17	253	234	186	249	236
1	171	54	76	203	55	63	81	1	171	144	142	250	140	242	81	81
2	4	171	82	171	71	148	251	21	204	235	1	49	136	49	189	19
case #2 ($x0$=0.2and k=2) and the number of sessions =3.																
0	247	190	1	1	9	111	14	1	71	49	33	246	76	30	186	31
1	16	193	31	115	222	24	90	17	93	106	103	64	145	235	89	30
2	35	196	1	211	189	63	19	1	171	16	16	16	81	55	41	72
case #3 ($x0$=0.3 and k=3) and the number of sessions =3.																
0	26	224	186	195	171	40	173	159	24	106	84	226	94	210	51	151
1	1	1	145	249	222	3	73	160	65	34	234	1	147	71	106	106
2	204	97	186	66	76	230	106	1	19	21	249	123	154	137	135	106

C. Results of Encryption/Decryption Lightweight Stage

The suggested system's second stage is encryption and decryption to use the lightweight algorithm. this stage is solely performed on the drone's side, where the proposed system implementation on salsa 20 is lightweight and applied to two types of data (video and text), The video that was recorded by the drone and encryption payload data and sent to GCS via NTRIP, so Table 2. Illustrated samples of video (MP4 Video type and 1280×720 Dimensions)with the different attributes recorded by drone through session fly.

Table 2. Sample of Video

No	1	2	3	4
Video size	11.2MB	14.3MB	5.68MB	4.03MB
Video length in a second	00:00:40	00:00:49	00:00:22	00:00:30

The proposed system firstly converts video to frames image see section 4.2, where take one sample image acquired from each sample. In Table

3. The results show the success of the salsa20 encryption algorithm in encoding the image, meaning that the cipher image is unintelligible and has no effect on the main image or the dispersed pixels of initial test images and cipher images. The results indicate that using salsa20 encryption techniques to cipher histogram images seems to improve NTRIP security, preventing unauthorized users from acquiring the original image before encoding.

Table 3. Image Encryption by salsa20 Lightweight Algorithm

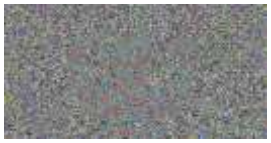
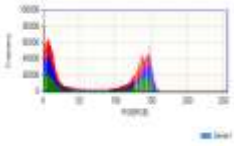
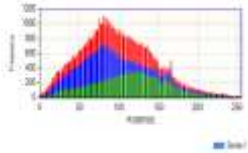
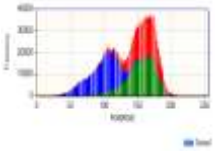
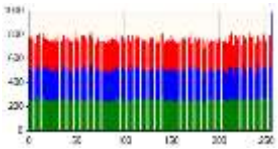
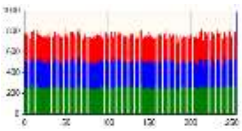
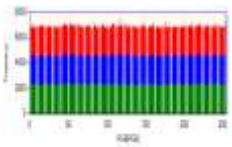
No.	1	2	3
Original Image			
Cipher Image			
Histogram Original Image			
Histogram Cipher Image			

Table 4 shows the results of error-sensitivity based metrics used between original and cipher images using the salsa20 algorithm. Mean Square Error (MSE) using equation (2), Peak Signal to Noise (PSNR) using equation (3), Normalized Cross Correlation (4), and Average Different (AD) using equation (5) are the error sensitivity metrics. The best MSE values are 9088.654987, the best PSNR values are 8.587654332, the best NCC values are 0.545689222, and the best AD values are 83.13457418. Figure 9 illustrates that the salsa20 cipher image of various sizes has a quick execution time. Table 4 and Figure 9 have been explained below respectively.

Table 4. Results of Error Sensitivity Based Metrics

Error Sensitivity Based Metrics		Image	Salsa20
1	MSE	Im1	4405.669205
		Im2	9088.654987
		Im3	6642.013472
		Im4	5156.547845
		Im5	5142.507929
2	PSNR	lm1	11.69068475
		Im2	8.587654332
		Im3	9.907806087
		Im4	11.00721309
		im5	11.01905391
3	NCC	Im1	0.629153028
		Im2	0.560039232
		Im3	0.545689222
		Im4	0.594587557
		im5	0.558858086
4	AD	Im1	65.42722703
		Im2	76.50540223
		Im3	83.13457418
		Im4	69.21349338
		im5	69.26755816

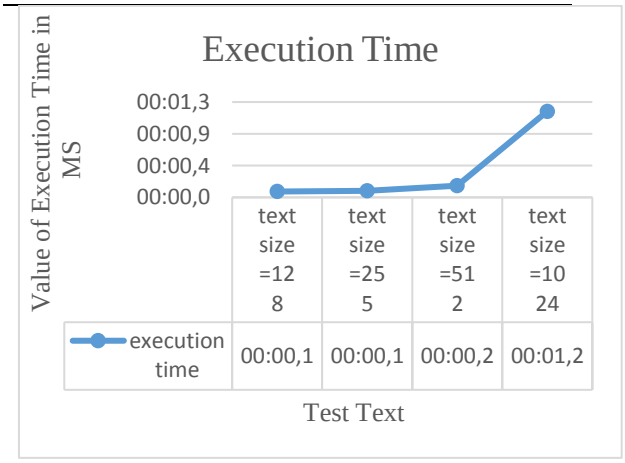


Fig. 9. Cipher Test Execution Time in MS Various-sized text, Salsa20 Lightweight was used in this work.

D. Performance Analysis

The suggested methods are compared to three state-of-the-art methods [30], [31], [4] which can be used to safeguard communication in a UAV network in this section. In terms of execution time, the comparison is made.

Table 5. Based on execution time, compare the proposed method to the previous one.

No	Reference	Security Methods	Execution Time
1	Jongho Won et al. [30]	eCLSC-TKEM	9.25 s
2	Christian Bunse, et. al. [31]	e Frequency Hopping Spread Spectrum (FHSS)	0.6 s
3	Hani M. Ismael et. al. [4]	HIGHT lightweight algorithm	1.4 msec
4	Our proposed method	Salsa20 algorithm	1.0 msec

VI. CONCLUSION

Although unmanned aerial vehicles (UAVs) play an important role in many application areas, security issues limit their ability to supply the desired solution. The security and privacy of unmanned aerial vehicles (UAVs) should be a top focus, especially in military scenarios. To solve the security issues, we proposed authentication and securing drone communications using salsa20 lightweight encryption algorithms. The proposed secured drone communications include three stages. The suggested system's initial stage is registration, while the second stage uses the salsa20 lightweight algorithm for encryption and decryption. The last stage in the suggested system is the authentication stage to check the authentication of the drone before receiving a massage. Results of the proposed system are done on two different types of data (colored image and text) with different sizes. Based on error sensitivity metrics the stream cipher salsa20 algorithm on ciphering colored images achieves higher results, The most important values of MSE = 9088.654987, the most important values of PSNR = 8.587654332, the most important values of NCC = 0.545689222, finally the best values of AD =83.13457418. In terms of speed salsa20 algorithm is faster in execution time in the case of ciphering colored images and text.

CONFLICT OF INTEREST

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

REFERENCE

[1] M. Moshref-Javadi and M. Winkenbach, "Applications and Research avenues for drone-based models in logistics: A classification and review," *Expert Syst. Appl.*, vol. 177, p. 114854, 2021, doi: 10.1016/j.eswa.2021.114854.

[2] P. Foehn *et al.*, "Alphapilot: Autonomous drone racing," *Auton. Robots*, vol. 46, no. 1, pp. 307–320, 2022.

[3] B. Sah, R. Gupta, and D. Bani-Hani, "Analysis of barriers to implement drone logistics," *Int. J. Logist. Res. Appl.*, vol. 24, no. 6, pp. 531–550, 2021.

[4] H. M. Ismael, "Authentication and encryption drone communication by using HIGHT lightweight algorithm," *Turkish J. Comput. Math. Educ.*, vol. 12, no. 11, pp. 5891–5908, 2021.

[5] J.-P. Yaacoub, H. Noura, O. Salman, and A. Chehab, "Security analysis of drones systems: Attacks, limitations, and recommendations," *Internet of Things*, vol. 11, p. 100218, 2020.

[6] H. P. D. Nguyen and D. D. Nguyen, "Drone application in smart cities: The general overview of security vulnerabilities and countermeasures for data communication," *Dev. Futur. Internet Drones Insights, Trends Road Ahead*, pp. 185–210, 2021.

[7] A. Yazdinejad, R. M. Parizi, A. Dehghantanha, H. Karimipour, G. Srivastava, and M. Aledhari, "Enabling drones in the internet of things with decentralized blockchain-based security," *IEEE Internet Things J.*, vol. 8, no. 8, pp. 6406–6415, 2020.

[8] S. Aggarwal, M. Shojafar, N. Kumar, and M. Conti, "A new secure data dissemination model in internet of drones," in *ICC 2019-2019 IEEE international conference on communications (ICC)*, 2019, pp. 1–6.

[9] Z. Lv, "The security of Internet of drones," *Comput. Commun.*, vol. 148, pp. 208–214, 2019.

[10] J. Shahmoradi, E. Talebi, P. Roghanchi, and M. Hassanalian, "A comprehensive review of applications of drone technology in the mining industry," *Drones*, vol. 4, no. 3, p. 34, 2020.

[11] P. Zhu, L. Wen, X. Bian, H. Ling, and Q. Hu, "Vision meets drones: A challenge," *arXiv Prepr. arXiv1804.07437*, 2018.

[12] P. Getsov, B. Wang, and D. Zafirov, "Precision drones–today and tomorrow," *Исследования Земли из Космоса*, no. 1, pp. 84–91, 2019.

[13] S. B. Sadkhan and A. O. Salman, "A survey on lightweight-cryptography status and future challenges," in *2018 International Conference on Advance of Sustainable Engineering and its Application (ICASEA)*, 2018, pp. 105–108.

[14] S.-L. Peng, S. Pal, and L. Huang, *Principles of internet of things (IoT) ecosystem: Insight paradigm*. Springer, 2020.

[15] N. A. Gunathilake, W. J. Buchanan, and R. Asif, "Next generation lightweight cryptography for smart IoT devices:: implementation, challenges and applications," in *2019 IEEE 5th World Forum on Internet of Things (WF-IoT)*, 2019, pp. 707–710.

[16] D. K. Lam, V. T. D. Le, and T. H. Tran, "Efficient Architectures for Full Hardware Scrypt-Based Block Hashing System," *Electronics*, vol. 11, no. 7, p. 1068, 2022.

[17] A. H. Fadel, R. S. Hameed, J. N. Hasoon, S. A. Mostafa, and B. A. Khalaf, "A light-weight ESalsa20 Ciphering based on 1D logistic and chebyshev chaotic maps," *Solid State Technol.*, vol. 63, no. 1, pp. 1078–1093, 2020.

[18] A. Jain and N. Rajpal, "A robust image encryption algorithm resistant to attacks using DNA and chaotic logistic maps," *Multimed. Tools Appl.*, vol. 75, no. 10, pp. 5455–5472, 2016.

[19] Z. A. Abduljabbar *et al.*, "Provably Secure and Fast Color Image Encryption Algorithm Based on S-Boxes and Hyperchaotic Map," *IEEE Access*, vol. 10, pp. 26257–26270, 2022.

[20] C. Pak and L. Huang, "A new color image encryption using combination of the 1D chaotic map," *Signal Processing*, vol. 138, pp. 129–137, 2017.

[21] M. Almazrooie, A. Samsudin, and M. M. Singh, "Improving the diffusion of the stream cipher salsa20 by employing a chaotic logistic map," *J. Inf. Process. Syst.*, vol. 11, no. 2, pp. 310–324, 2015.

[22] Y. A. Y. Al-Najjar and D. C. Soong, "Comparison of image quality assessment: PSNR, HVS, SSIM, UIQI," *Int. J. Sci. Eng. Res*, vol. 3, no. 8, pp. 1–5, 2012.

[23] Y. Liu, J. Tang, and T. Xie, "Cryptanalyzing a RGB image encryption algorithm based on DNA encoding and chaos map," *Opt. Laser Technol.*, vol. 60, pp. 111–115, 2014.

[24] K. Domin, I. Symeonidis, and E. Marin, "Security analysis of the drone communication protocol: Fuzzing the MAVLink protocol," 2016.

[25] S. A. Thomas and S. Gharge, "Halftone visual cryptography for grayscale images using error diffusion and direct binary search," in *2018 2nd International Conference on Trends in Electronics and Informatics (ICOEI)*, 2018, pp. 1091–1096.

[26] K. Aoki, T. Ichikawa, and M. Kanda, "A 128-Bit Block Cipher Suitable for Multiple Platforms—Design and Analysis//International Workshop on Selected Areas in Cryptography." Springer Berlin Heidelberg, 2000.

[27] M. Yahuza *et al.*, "Internet of drones security and privacy issues: Taxonomy and open challenges," *IEEE Access*, vol. 9, pp. 57243–57270, 2021.

[28] N. M. Sahib, A. H. Fadel, and N. S. Ahmed, "Improved RC4 algorithm based on multi-chaotic maps," *Res. J. Appl. Sci. Eng. Technol.*, vol. 15, no. 1, pp. 1–6, 2018.

[29] S. Ji and K. Wan, "Adaptive Modular Exponentiation Methods vs Python's Power Function," *arXiv Prepr. arXiv1707.01898*, 2017.

[30] J. Won, S.-H. Seo, and E. Bertino, "A secure communication protocol for drones and smart objects," in *Proceedings of the 10th ACM symposium on information, computer and communications security*, 2015, pp. 249–260.

[31] C. Bunse and S. Plotz, "Security analysis of drone communication protocols," in *International Symposium on Engineering Secure Software and Systems*, 2018, pp. 96–107.

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

All authors are involved in conceptualization, formal analysis, simulation, writing up, and editing.

Sources of Funding for Research Presented in a Scientific Article or Scientific Article Itself

This research article has no funded or granted project.