Computers, Materials & Continua
DOI:10.32604/cmc.2022.019419
Article

Tech Science Press

A Secure Communication Protocol for Unmanned Aerial Vehicles

Navid Ali Khan[1], N. Z. Jhanjhi[1,*], Sarfraz Nawaz Brohi[2], Abdulwahab Ali Almazroi[3] and Abdulaleem Ali Almazroi[4]

[1]School of Computer Science and Engineering (SCE), Taylor's University Lake-side Campus, Subang Jaya, 47500, Malaysia
[2]School of Information and Technology, Monash University, Malaysia
[3]University of Jeddah, College of Computing and Information Technology at Khulais, Department of Information Technology, Jeddah, Saudi Arabia
[4]Department of Computer Science, Rafha Community College, Northern Border University, Arar, 91431, Saudi Arabia
[*]Corresponding Author: N. Z. Jhanjhi. Email: noorzaman.jhanjhi@taylors.edu.my
Received: 13 April 2021; Accepted: 18 May 2021

Abstract: Mavlink is a lightweight and most widely used open-source communication protocol used for Unmanned Aerial Vehicles. Multiple UAVs and autopilot systems support it, and it provides bi-directional communication between the UAV and Ground Control Station. The communications contain critical information about the UAV status and basic control commands sent from GCS to UAV and UAV to GCS. In order to increase the transfer speed and efficiency, the Mavlink does not encrypt the messages. As a result, the protocol is vulnerable to various security attacks such as Eavesdropping, GPS Spoofing, and DDoS. In this study, we tackle the problem and secure the Mavlink communication protocol. By leveraging the Mavlink packet's vulnerabilities, this research work introduces an experiment in which, first, the Mavlink packets are compromised in terms of security requirements based on our threat model. The results show that the protocol is insecure and the attacks carried out are successful. To overcome Mavlink security, an additional security layer is added to encrypt and secure the protocol. An encryption technique is proposed that makes the communication between the UAV and GCS secure. The results show that the Mavlink packets are encrypted using our technique without affecting the performance and efficiency. The results are validated in terms of transfer speed, performance, and efficiency compared to the literature solutions such as MAVSec and benchmarked with the original Mavlink protocol. Our achieved results have significant improvement over the literature and Mavlink in terms of security.

Keywords: Unmanned aerial vehicles; mavlink protocol; drones security; UAVs communication

1 Introduction

Unmanned Aerial Vehicles (UAVs) have become increasingly common in recent years. By expanding their military reach to commercial use, these unmanned aerial systems are being used worldwide [1]. An Unmanned Aerial Vehicle is a pilotless aircraft without any crew on

board [2,3]. They can operate completely autonomously [4,5] through autopilot, which means they can complete tasks without human interaction or can be non-autonomous with the help of remote control or a human pilot operating in runtime from a ground control station [6]. The most common use is either by the military [7–9] for battlefields or reconnaissance or by civilians for leisure and entertainment [10,11]. Following this, there are many other emerging applications of the UAVs such as agriculture [12,13], environmental protection [14], search and rescue [15], traffic monitoring [16], delivery [17], aerial mapping [18], aerial photography [19] and videography [20,21], fire detection [22].

Moreover, the tech giants like Facebook [23] and tesla are using UAVs to bring the internet to places around the world where there is no internet access [24]. The UAVs are best suited for "dull, dirty and dangerous" [25] missions, which means the situation where it is too difficult for a human to reach and operate. When a public communicating network gets crippled, UAVs will provide prompt disaster alerts and help speed up rescue or recovery operations. They can transport medical supplies to inaccessible areas. UAVs may be used to rapidly cover a wide area without endangering the workers' protection in the circumstances like toxic gas incursions, wildfires, or wild animal monitoring. The widest and important use of UAVs can be seen in military applications. Countries with advanced Unmanned Aerial Systems overpower their rivals prominently because of their greater stealth, reduced scale, and real-time capability in harsh environments and frontier surveillance [26]. Therefore, UAVs can be effectively used in various situations to combat terrorism without loss of life. The real-time monitoring activity would, in this sense, require multiple details gathered during UAV flight, such as a telemetric subsystem, payload subsystem, and commands to be transmitted from the ground station for UAV management and mission performance [27].

The Unmanned Aerial Vehicles are the components of an Unmanned Aerial System (UAS). The three main components of a UAS are a UAV, a Base Station/Ground Control Station, and the communication system that links them [28]. Various communication protocols are typically used to communicate between the UAV and the GCS, such as Mavlink [29], UAVCan [30], UranusLink [31]. Among these protocols, the Micro Aerial Vehicle Link (Mavlink) protocol is the most common and widely used communication protocol supported by a large number of UAVs and the Ground Control Station [32]. Lorenz Meier first developed this protocol in 2009 under GPL license [33]. Mavlink allows the UAV and GCS to communicate in both directions. The Ground Control Station sends instructions and controls messages to the UAV, and the UAV sends telemetry and other status information to the GCS [34]. UAVs are also linked over the internet using the Mavlink protocol [35].

The Mavlink protocol is supported by many UAVs and several autopilot systems such as Ardupilot [36] and PX4 [37]. These two open-source systems are the leading autopilots that can control any unmanned vehicles ranging from unmanned aerial vehicles to even unmanned submarines [36,38].

Mavlink is a lightweight networking protocol that is open-source and cross-platform. Mavlink 1.0 [33], Mavlink 2.0 [39], and a prototype version sMavlink are the three versions available. Mavlink 2.0 uses timestamped hash-based message authentication codes for integrity and authentication (HMAC).

Mavlink is structured as a Marshalling library, which means that the system states' messages and the commands it requires to run in a particular binary format are serialized (streams of

bytes) independent of the platform. Mavlink's binary serialization approach is lightweight and low overhead as compared to other serialization methods like XML and JSON.

The sMavlink draft version is a stable version that ensures confidentiality and integrity by using symmetric key authenticated encryption of relevant details [40]. To the best of our knowledge, the sMavlink is not implemented yet.

Furthermore, Mavlink messages are usually small and can be transmitted over a range of wireless networks, including Wi-Fi or even serial telemetric systems with low data rates, due to its Binary Serialization features. A double checksum verification guarantees message durability and accuracy in the packet header. The Mavlink protocol is the most widely used by its peers for communication between unmanned systems and ground control stations (GCS) due to these characteristics.

Despite being robust and most widely used, the Mavlink communication protocol lacks a subtle security mechanism, making it vulnerable to several attacks, such as Denial of services attacks (DDoS), Eavesdropping, and Man-in-the-middle attack [41,42]. These vulnerabilities are apparent because the Mavlink protocol does not encrypt the messages in communication. That means that the binary communication between the GCS and the UAV is happening over an unencrypted channel, making it an easy target for different security attacks. Thus, compromising the security of Unmanned Aerial Vehicles.

This work's main contribution is an additional security layer added to the Mavlink communication protocol to secure the binary directional communication between the UAV and GCS. Our research produced three algorithms. The other contributions and three algorithms are described below

I. Developed an algorithm to relaunch the captured Mavlink packets for attacks.
II. Developed an algorithm to retrieve meaningful information from captured Mavlink packets.
III. Developed an algorithm to encrypt the Mavlink packet to secure the communication.

(1) We performed and tested the experiment in a simulating environment using Ardupilot and Mission planner, which use the same Autopilot software as used in real UAVs.
(2) We secured the Mavlink protocol for communication between UAV and GCS without affecting performance and efficiency.

The rest of the paper is organized as follows. Section 1 presented the Introduction. Section 2 presents the literature review and related work. A detailed overview of the Mavlink protocol is given in Section 3. Section 4 describes the Security issue of the Mavlink protocol. In Section 5, the Mavlink protocol has been exploited in terms of security attacks and vulnerabilities. Two algorithms have been proposed in this section to exploit the Mavlink protocol vulnerabilities. Section 6 presents the proposed solution to secure the Mavlink protocol. The encryption algorithm and our security technique are illustrated in this section. Section 7 demonstrates the experimental results and our solution's benchmarking with the original Mavlink protocol in terms of performance and efficiency. Finally, the conclusion of the paper is presented.

2 Literature Review and Related Work

The threat against UAVs is often targeted at the Unmanned Aerial System. It can be any components from the three, the UAV, the Ground Control Station, or the communication link between the two [43]. In this study, the focus is on communication link attacks, as shown in Tab. 1.

Table 1: Communication link attacks on UAVs

Security objectives	System objectives	Attack methods
Confidentiality	Ground control station	Virus
		Malware
		Key loggers
		Trojans
	UAV	Hijacking
	Communication link	Eavesdropping
		Man-in-the-middle
Integrity	Communication link	Packet injection
		Replay attack
		Man-in-the-middle
		Message detection
Availability	GCS	Denial of services (DoS)
	UAV	Fuzzing
	Communication	Jamming
		Flooding
		Buffer overflow
		Denial of services

As Unmanned Systems has seen tremendous growth in recent years, therefore their security has become very crucial. Many researchers have contributed to this field, and much work has been done. The contributions can be mainly divided into two approaches 1) Hardware and 2) Software.

In order to protect the Mavlink protocol, many embedded and hardware security technologies have been implemented. In the work proposed in [44], the researchers used additional encrypted communication channels with Raspberry Pi's help to the UAV security issue. In this solution, the hardware has to communicate with GCS to regain control if an attack occurs. The downside of this approach is the time difference between the GCS and the Raspberry and the higher CPU consumption. Another drawback of the analysis is that it is just a theory that has yet to be tested on real UAVs. Our study implements the solution on a case study, and the results are given in Section 7.

In another study [45], the authors secure the communication between the GCS and UAV through a proposed AES protocol with hardware implementation. The main focus of this study is confidentiality and authentication. However, the given hardware solution harms the system's efficiency, CPU, and energy consumption because of the additional hardware weight.

On the other hand, contributing to the software solutions of the Mavlink protocol, the authors consider using Caesar cipher cryptography for data encryption and authentication of Mavlink messages between the ground station and the UAV [46]. One limitation of this study is that they didn't give the results in the study. Another drawback of their work is that they are sending the secret Key in plain text. In [47], with efficient symmetrical key encryption algorithms, four effective cryptographic solutions were applied to reduce the confidentiality vulnerabilities in the Mavlink Protocol. Rabbit stream cipher, Salsa20 stream cipher, and XXTEA stream cipher are the four algorithms that have been proposed. All of them can conveniently encrypt Mavlink messages while keeping GCS and UAV communications private. The research articles [46,48] use

the Caesar cryptography algorithm to encrypt Mavlink messages between GCS and UAV for cryptographic data purposes. However, in this solution, the hidden Key is sent to the UAV in plain text during the establishment process. It is effortless to find the Key at the time of capturing the packet. Thus, it is very easy to break its security. Moreover, there is no empirical evaluation of the study. Our research work implements the solution on a case study, and the results are presented below. In another study, another encryption RC5 is used to ensure that the communication is secure, but there are no details or validation of the experiment [49]. Our study secured the communication between UAV and GCS and analyzed and validated the performance with clear results. The research [50] suggests using the UAV's Private Key to add a digital signature to the data packing. In [51], another author proposed cryptographic encryption for authentication to ensure the integrity of data. However, both these studies are just proposed work. The author [47] conducted a vulnerability analysis and suggested a cryptographic algorithm to protect the Mavlink protocol without defining which algorithm to use. In [37], the authors proposed a solution called MAVSec to secure Mavlink communication. They compared four encryption algorithms, including AES-CBC, AES-CTR, RC4, and ChaCha20. Based on their result, ChaCha20 seems to be giving good results compared to others in terms of performance. However, in their proposed method, the encryption is only applied to the payload messages. The rest of the packet is the same. In our solution, we provided an extra layer of security that secures the whole packet. Several other studies have been carried out to secure the Mavlink communication protocol. Still, most of the studies are just proposed solutions or are in their early development.

3 Overview of the Mavlink Protocol (Mavlink System Architecture)

The Mavlink protocol specifies the framework for message composition and how to serialize messages on an application layer. The serialization process involves converting into a later stored or distributed format of a data structure or object state. After serialization, these messages are transferred to the lower layers, i.e., the transport layer and physical layer, to be sent over the network. The lightweight construction allows it to accommodate a number of transport layers and media. The Mavlink protocol can be transmitted over sub-GHz frequencies like 433, 868, and 915 MHz using Wi-Fi, TCP/IP, or low-bandwidth serial telemetry networks [52].

The second option is to use normally a Wi-Fi or Ethernet network interface to stream Mavlink messages through IP networks. In the transport layer, the Mavlink autopilot accepts both UDP and TCP links between the ground control station and the UAV, depending on the application's configuration. A connection between the client and the server is not needed for the datagram protocol UDP [53]. Therefore, it is unreliable in terms of message delivery. The advantage is that it provides a fast, light alternative weight for streaming real-time, loss-tolerant communication. In contrast to UDP, TCP is a connection-oriented protocol, which ensures it has a mechanism for acknowledging that the request has been sent [54]. This means that TCP is reliable in terms of communication. Depending on the requirements, the user has to choose whether to use UPD or TCP protocol.

The communication between the UAV and Ground Station occurs through binary serialized messages. Since the communication is bidirectional, the message's serialization and deserialization take place at both the sender and recipient ends. In comparison to other serialization approaches, Mavlink serialization uses fewer transmission messages and is significantly lighter. The Mavlink protocol has two available versions Mavlink 1.0 and Mavlink 2.0 [52]. There's another version of Mavlink called sMavlink. To the best of our knowledge, the sMavlink is not implemented yet.

4 Security Issues of Mavlink Protocol

The research and development in Unmanned Systems is relatively a new area, and still, a lot of research and development work is in progress. In parallel, the hackers and attackers find it an opportunity to explore new vulnerabilities and compromise these systems' security with various intentions. To address security issues and challenges, many researchers have contributed to Unmanned Systems security at different levels. One of the issues with those solutions is that they are only in the early stages of implementation and either or only proposed work. Before providing the solution to exploit the Mavlink protocol's vulnerabilities, we need to understand the security challenges. In what follows, the security challenges of the Mavlink protocol can be divided into 1) Security Requirements, 2) Security Threats/Attacks. This will help the practitioners and researchers in the future to develop security frameworks and threat models for Unmanned Aerial Vehicles.

4.1 Security Requirements

Overall, there has been much research done in terms of unmanned aerial systems security, but less work has been done on communication level security, particularly on the Mavlink protocol. A medical term is best suited for the security requirements as it says, "prevention is better than cure." To avoid security threats and attacks, it is most important to understand the security requirements and avoid these unwanted situations. The Mavlink's security requirements are summarized as confidentiality, integrity, availability, authentication, non-repudiation, authorization, and privacy [3] to secure the communication between the UAV and GCS and avoid threats. Fig. 1 below presents the Mavlink security requirements.

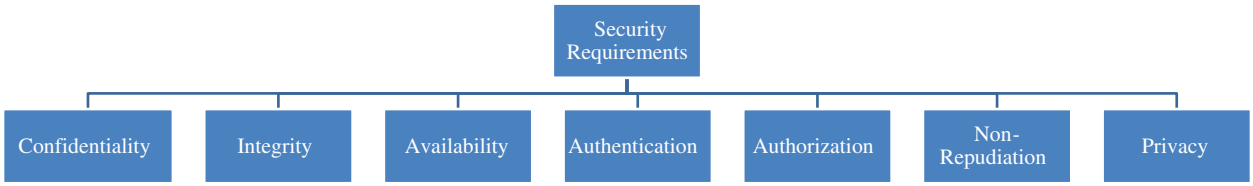

Figure 1: Mavlink security requirements

4.2 Security Threats

The connectivity between the UAV and GCS occurs through a wireless channel with the communication protocol's help. In the case of the Mavlink protocol, this communication is vulnerable because the Mavlink protocol does not have standard security procedures. The only security check is that it checks if the packet is authentic and comes from an authentic source. The rest of the security requirements, such as confidentiality, is not natively available. The Mavlink does not have a subtle security mechanism and does not encrypt the messages. That means that the UAV and Ground Control Station communication is not secure and can be compromised very easily. Any hacker or attacker with an appropriate transmitter device can intercept the communication and communicate with the UAV. The intruder can use this vulnerability for their intended purpose, such as inject false commands into an existing mission or hijack the UAV completely. Further, these attacks are classified in terms of their outcome as follows. The classification is given in Tab. 2.

Table 2: Security threats/attacks against mavlink protocol [3,55]

Security requirement	Threats/attacks	Mitigations
Confidentiality and privacy	Man-in-the-middle Eavesdropping Identity spoofing Hijacking Unauthorized access Interception	Datalink encryption
Integrity	Packet injection Man-in-the-middle Fabrication Message deletion Message modification Replay attack	Hash MAC (message authentication code) Authentication
Availability	Command and control Jamming Routing attack Denial of service Flooding	Authentication
Authenticity	GCS Spoofing Fabrication	Authentication

Based on the above security threats, we present our threat model and exploit the vulnerabilities of the Mavlink communication protocol. The threat model consists of two steps 1) to exploit the Mavlink packet and use it for active attacks 2) exploit the Mavlink packets and later use it for passive attacks. For this purpose, two algorithms are developed and presented in the next section. The threat model is illustrated below in Fig. 2. As shown in Tab. 1, our focus here is to target the UAV against the communication link attacks and hijack the UAV. Based on our proposed algorithm to exploit the Mavlink vulnerabilities in our experiment. First, we carried out a Man-in-the-middle attack to capture the packets. When the packets are captured, Algorithm 1 relaunches the captured Mavlink packets for a replay attack. This can be used for two purposes: 1) relaunch the packets for a replay attack or an eavesdropping attack if it is an ongoing mission. If the intention here is to inject false data, a false injection attack can be carried out too by inserting false data into the captured packets. Our experiment, based on Algorithm 1, hijacked the UAV and took full control of it. Similarly, Algorithm 2 is basically developed to understand the captured packets communication between the UAV and GCS. It can be launched for passive attacks.

5 Exploiting the Mavlink Protocol

The experiment is carried out in a simulated environment using ArduPilot Software in the loop (SITL) and a simulating UAV. The Adrupilot SITL uses the same autopilot which is used in a real UAV. It replicates the real UAV in a simulated environment, and it also can operate a plan or a land rover without using any hardware. For the Ground Control Station, Mission Planner is selected, so at this moment, when we open Mission Planner, it automatically connects to the

UAV via Mavlink protocol. If it does not connect automatically, it can be connected manually by clicking the connect button on the top right corner of the Mission Planner application. After Successful execution, the Unmanned Aerial Vehicle can be seen on the Mission Planner map as well. Here we can define a new mission, load an old mission and perform some other required tasks required for the mission.

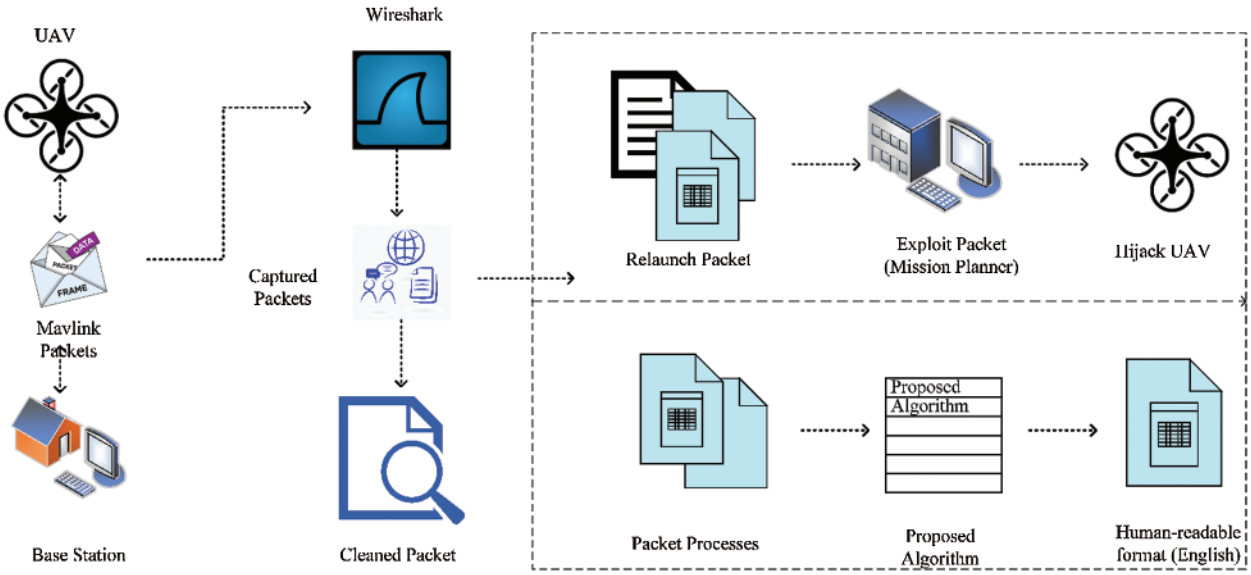

Figure 2: Threat model to exploit the Mavlink vulnerabilities

5.1 Capturing the Mavlink Packets

The Mavlink packets are captured by intruders using specific transmitters. Since we are simulating, the packets are captured on the local Wi-Fi network using the packet capturing tool Wireshark. We filtered the port numbers in Wireshark to get only our desired port number for Mavlink. It is effortless for a knowledgeable person to recognize the Mavlink packet structure. Once the packet structure is identified, it is easy to find the port from where the packets are coming and then filter only that port to get information of Mavlink packets.

The detailed structure of the Mavlink packet is shown in Fig. 3 above captured through Wireshark. The information displayed here is in binary and hex form. The packet is saved in a text file with proper formatting. In the next step, Algorithm 1 is developed to retrieve the useful information from this packet and relaunch it for an attack to get unauthorized access to the UAV.

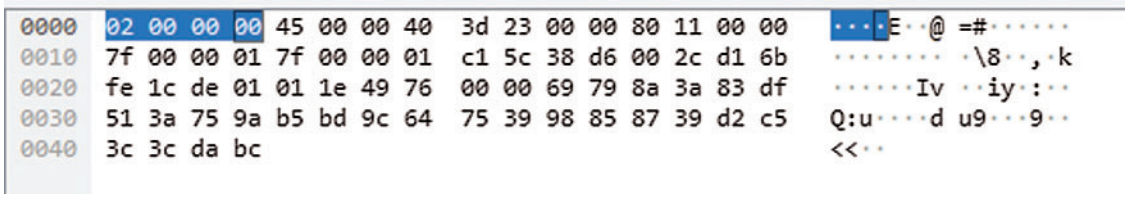

Figure 3: Captured mavlink protocol packet structure

Algorithm 1: Launching the captured packet for attacks

Input: *Captured_Packet*
Output: *Hijack_Drone*
Procedure (*Exploiting_Mavlink*)
1 $D_0 \leftarrow Read\,(Caputred\_Packets)$
2 $D_1 \leftarrow Read\_Bytes\,(D_0)$
3 $Socket \leftarrow Define\,(Port_{num})$
4 $While\,(i \leftarrow Read\,(D_1))$
5 $Packet \leftarrow Get\,(Length, add, port)$
6 $Send\,(Packet) \rightarrow Port\,(Socket)$
7 $GCS\,(Mission\_Planner) \leftarrow Received\,(Packets)$
8 *End procedure*

Algorithm 1 is implemented using Java code. When the code is run, it gets the data from the Mavlink packet that we captured via Wireshark, stored in a text file. It reads the buffers and the start sending the data on a defined port number (14450 in our case). As long as it is reading the data from the packet, it keeps sending the data. Since we are using a mission planner on our side, the Mission Planner received the data via the specified port number. All the UAV information from which the packet was captured is now visible, such as mission data and GPS location. If it's an ongoing mission, the UAV can completely hijack from Mission Planner as now it has control of the UAV. Moreover, any other intended attack can be launched, such as eavesdropping, GPS Spoofing, False mission data injection.

5.2 Converting the Packet into a Human-Readable Form

The next step is to exploit the Mavlink protocol's vulnerability that it is not secure and the data transmitted is not encrypted. For this purpose, we developed Algorithm 2, which converts the information captured by the Wireshark to a human-readable format in plain text and retrieves the meaningful information from that. This algorithm is also implemented using Java Coding.

Algorithm 2: Getting secret information in plain text

Input: *Captured_Packets*
Output: *Plain_Text*
Procedure (*Exploiting_Mavlink*)
1 *CreateDroneObject*
2 *DefineMavlinkMessageHandler*
3 $D_0 \leftarrow read\,(Datablock\_Packets)$
4 $While\,(Connection! = 0)$
5 $Packet \leftarrow Get\,(length, addr, port)$
6 $Response\,(Mavlink\_Library) \leftarrow read\,(Mavlink\_Messages)$
7 $For\,(i \leftarrow Read\,(Packet))$
8 $D_0 \leftarrow Parse\,(Packet) + 0x00ff$
9 $D_1 \leftarrow Parse\,(Hex\_Char)$
10 $DroneObject \leftarrow Ready\,(D_0, D_1)$

(Continued)

```
11          if (Packet! = Null)
12              Unpack (Packet)
13              Received_Data (Binary, Hex)
14              Plain_Text ← Convert (Binary, Hex)
15              Sec_Inf ← Get (GPS, payload, alt, etc.,)
16  End procedure
```

In this algorithm, first, a connection is established, and then a Mavlink message handler is defined, which checks the Mavlink messages and understands what type of Mavlink message it is handling. Then the connection is checked if it is connected to the drone object. As long as the connection is built, first, it will read the data block and save it. Then we have our library, which understands the Mavlink message. After this, the data is parsed, and the hex value 0x00ff is added, which checks which type of Mavlink packet it is, the actual hex value is received. Then we parse the data from the derived hex characters. An object is made for the Mavlink packet and checks if the packet is not null (real packet). If the packet is real, it unpacks the Mavlink packet and fetches the data into human-readable form. All attributes can be fetched from here, such as sensors data, roll, pitch, radio frequency, GPS, etc.

6 Proposed Solution

In this section, we propose our solution for the runtime security of the Mavlink protocol for an ongoing mission between UAV and GCS. Our approach is based on a cryptographic mechanism and our mapping technique. A security layer is added to the original Mavlink protocol, and the overall proposed model is given in Fig. 4. We encrypt the information in the Mavlink packet and make sure that when the packet is captured through Wireshark or any transmitter device, 1) it can't be relaunched to take control of the UAV, i.e., the captured packet is useless for the intruder. 2) to encrypt the packets so that even if an intruder captures the packet, he/she still would not be able to get the meaningful information from this packet. We take a case study for our solution. As stated earlier, the experiment is carried out in a simulated environment. We assume that the ongoing mission is a minimum of ten minutes and a maximum of 3 h, mostly the case in terms of the UAV's civilian applications. The UAV and GCS are connected through Wi-Fi.

First, we apply custom mapping to the Mavlink packet, which replaces the character's ASCII. The data is basically bytes that are in binary and hex form. It will give us an ASCII string that is composed of random characters. This can be reverted only with the same mapping technique. Our approach introduces the concept of lists on both UAV and GCS sides, as shown in Fig. 5. The list has two columns, a serial number, and a key. A serial number is a representation number from the list to match the Key, while the key column contains the actual Key for the Caesar cipher through which the message will be encrypted. For instance, if the Key against serial number 2 is selected from the UAV side, it means that the GCS should decrypt the same serial number 2 to decrypt the message. In communication, instead of sharing the Key, the serial number is sent. So that even if the packet is captured, the intruder won't get the Key but the serial number, which is meaningless to decrypt the cipher. The serial number is added to the start of the packet, which takes four bytes. Once the serial number is added, the Caesar cipher encryption is applied to the packet based on the Key against the selected serial number in the list. It is thenconverted into bytes and sent to the GCS.

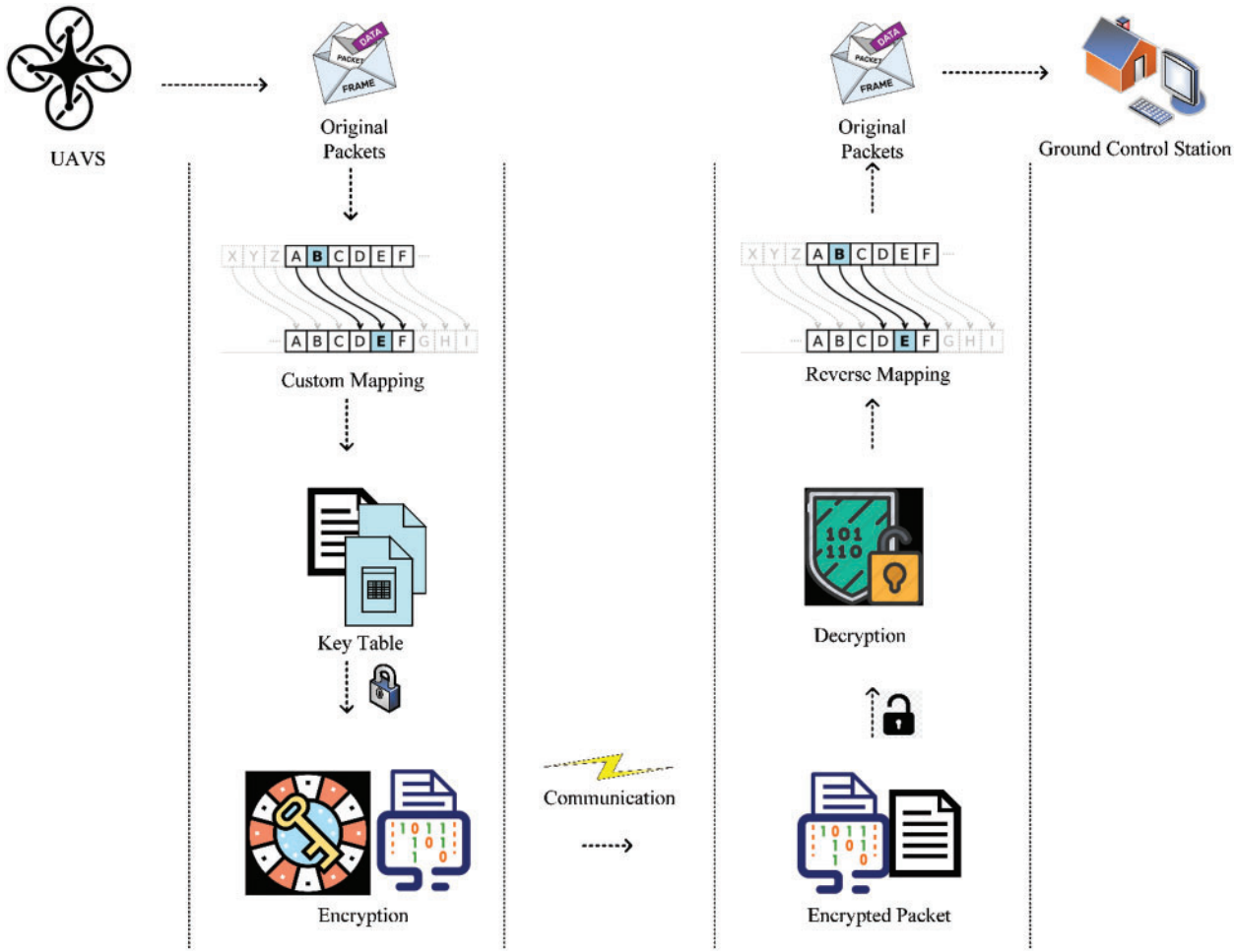

Figure 4: Proposed solution to enhance and secure the mavlink communication protocol

As the UAV receives the data, the first four bytes of the packets are taken as they contain the serial number. The serial number is matched with the list on the UAV side. Based on the serial number, the Key is identified, and a reverse Caesar cipher is applied to decrypt the message. When the data is decrypted, we get a character string that is still encrypted with our custom mapping and can be reverted only with our mapping reverting technique. We have also integrated the encryption technique in the Mission Planner to ensure safe communication between the autopilot of the SITL UAV and the Ground Control Station to decipher the obtained packet and to retrieve the original Mavlink message. An algorithm is developed to carry out this encryption process. The coding is implemented in Java. The pseudo-code of algorithm number 3 is given below in two steps—Algorithm 3 for Encrypting and sending the packets and Algorithm 4 for receiving and decrypting the packets.

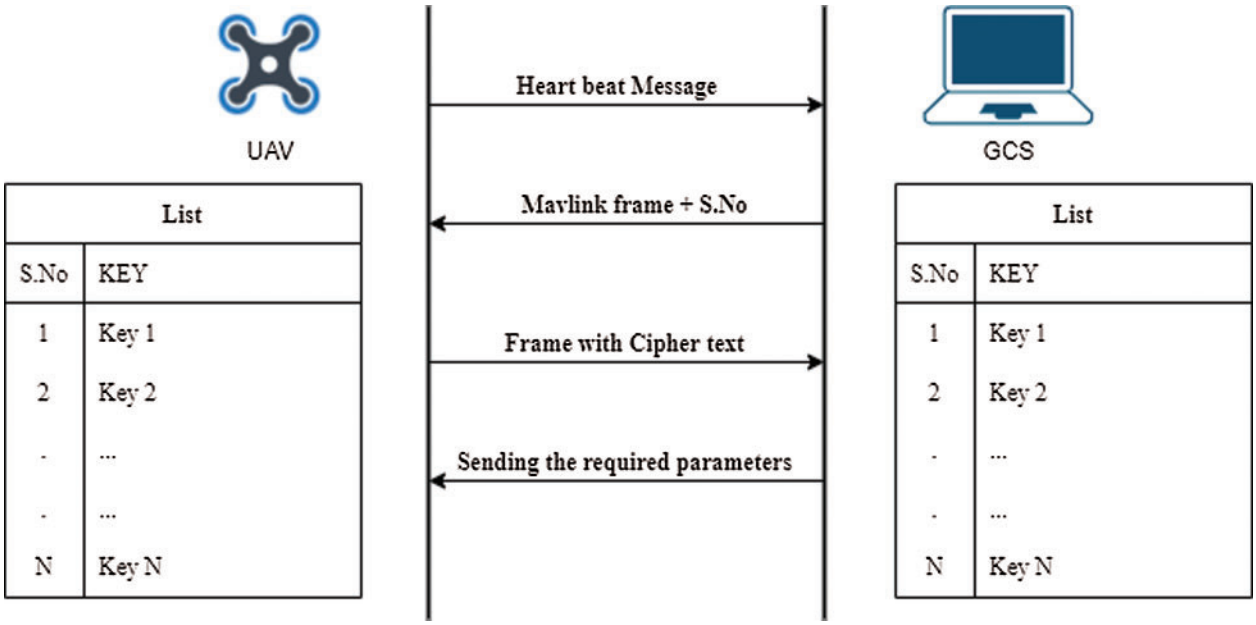

Figure 5: Lists of serial number and secret key for encryption

Algorithm 3: Encrypting the mavlink packet

Input: *Mavlink_Packets*
Output: *Encrypted_Mavlink_Packets*

Procedure (*Encryption_Mavlink*)
CreateDroneObject
$D_0 \leftarrow Packet\_Data$
$inveral \leftarrow IV$
if ($Packet\_Count = Interval$)
  Reset (*counter*)
  Swap (*Key_List*)
PickRandomIndex
$Fetch\_Key \leftarrow Rand\,(Index)$
$Hex\_String \leftarrow Convert\,(Fetch\_Key)$
For ($n \leftarrow Each\_Char$)
  if ($Char == Alphabet$)
    $Caesarcipher\_Encrypt\,(Char, Key)$
  Else if ($char == Num$)
    $Caesarcipher\_Encrypt\,(char, key + 2)$
  Switch
    $Custom\_Mapping\,(Char) \rightarrow Encrypted\_String$
  Send Encrypted Packets
End procedure

Algorithm 4: Receiving and decrypting packet

Input: *EncryptedMavlink_Packets*
Output: *Decrypted_Mavlink_Packets*
Procedure (*Decryption_mavlink*)
Create Drone Object
$D_0 \leftarrow Packet\_Data$
$inveral \leftarrow IV$
if ($Packet\_Count = Interval$)
  Reset (*Counter*)
  Swap (*Key_List*)
$S\_number \leftarrow Fetch\,(Packet)$ //*Fetch the serial number form first* 4 *bytes*
$Fetch\_Key \leftarrow List\,(S\_Number)$
Remove first 4 *Bytes*
For ($n \leftarrow Each\_Char$)
  Switch
    $Custom\_Reverse\_Mapping\,(Char) \rightarrow Decrypted\_String$
  if ($char == Alphabet$)
    $Caesarcipher\_Decrypt\,(Char, Key)$
  Else if ($Char == Num$)
    $Caesarcipher\_Decrypt\,(Char, Key - 2)$
  Original Mavlink Packet
End procedure

While exchanging the UAV and GCS data, the serial numbers are selected randomly from the table for every single request. It allows to change Key for every single request and encrypt every packet with a different key. This means that the next Mavlink packet will not send the same serial number but instead a new serial number with a new key against it to encrypt the message. When a hundred requests are completed, the serial numbers and the Key in the list are shuffled randomly in parallel on both sides so that the lists are similar on both sides. In case there at one end, the list is not updated after the iteration and sent to the other side, then the communication cannot happen, and the UAV/GCS will be considered unauthorized.

7 Performance Evaluation and Benchmarking

This section provides an exhaustive analysis of the Mavlink protocol's efficiency integrated with our encryption technique concerning the protocol's security. In addition, the output is evaluated in terms of resource use, such as CPU processing and memory consumption rate. We benchmark our proposed technique with the original insecure Mavlink protocol. In contrast, our technique makes sure the communication between the UAV and GCS is secured and the information shared is not vulnerable. Thus, our technique secures the Mavlink packet without affecting its performance and efficiency.

The experiment is to run a computer with an Intel 2.6 GHz Core i7 CPU; the memory or RAM is 8 GB. The operating system is Microsoft Windows 10 (64-bit) with Mission Planner 1.3.74, Ardupilot version 3.2.1, and UAV copter in SITL. The UAV is connected using UDP port number 14550.

7.1 Security

The experiment results show that the packet is encrypted with our encryption technique. The information shared cannot be retrieved when the packet is captured. To test this, we analyzed the communication between the UAV and GCS through our Mavlink encryption technique. First, the secured Mavlink packets are captured through Wireshark. Then the captured packets are sent based on Algorithm 1 to launch for attacks. The results show that when the packets are sent, the mission planner does not recognize the packets as they cannot retrieve the packet's data. Furthermore, based on Algorithm 2, it is tested whether the packets can be converted into a human-readable format or no; the results are negative. This means that the packets are useless to replicate for launching attacks as they are encrypted to secure the communication between UAV and GCS.

7.2 Transfer Speed (Packets Count)

The experimental results show that our approach sends almost the same number of packets as the original Mavlink does per second time. There's just a slight difference of one or two packets up and down which is negligible. The number of packets sent per second time by the original Mavlink protocol and our approach are given in Fig. 6.

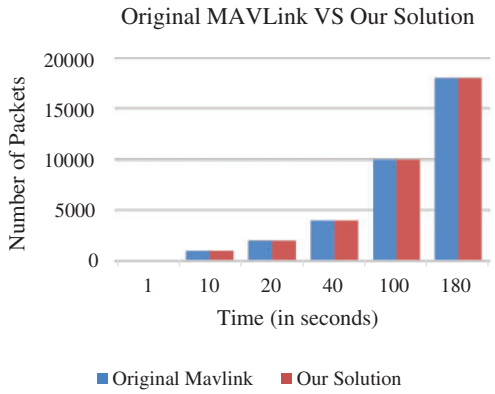

Figure 6: Number of packets transmission (in seconds)

7.3 Memory Consumption

Another important parameter for performance evaluation is memory consumptions. The results show that our technique's memory consumption is almost the same as the original Mavlink packet, as presented in in Fig. 7.

7.4 CPU Usage

Another important parameter for performance evaluation is memory consumptions. The results show that our technique's memory consumption is almost the same as the original Mavlink packet, as presented in in Fig. 8.

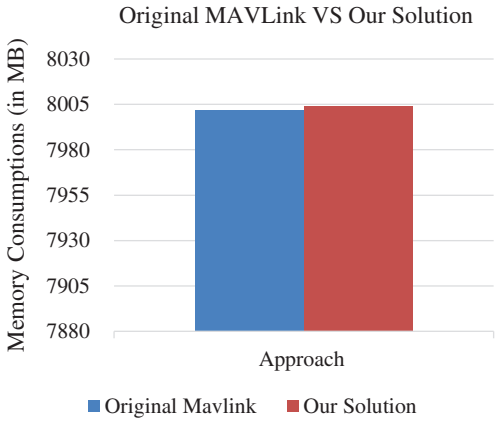

Figure 7: Memory consumptions (in MBs)

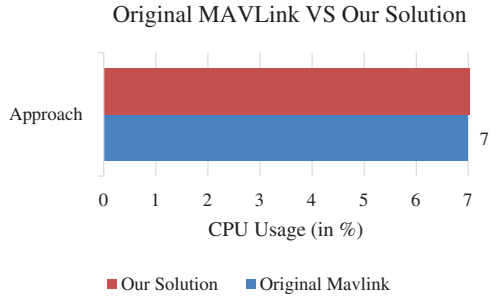

Figure 8: CPU processing (in percentage)

8 Conclusion

In this research work, a new approach is proposed and applied to secure the Mavlink communication protocol. The technique is based on a cryptographic encryption algorithm and custom mapping. An additional security layer is added to the Mavlink communication protocol to secure the whole packet. A new concept of lists is introduced, which sends a serial number instead of sending the secret Key for encryption. The Key against the serial number of both sides is matched, and the messages are encrypted and decrypted. The results are carried out in simulating the environment using virtual UAV via Ardupilot SITL, which uses the same autopilot as the real Planes and UAVs. The result shows that our technique makes the communication secure without affecting the original protocol's performance and efficiency. The proposed solution is compared with the existing literature, such as MAVsec and benchmarked with the original insecure Mavlink protocol to validate the results in terms of transfer speed, performance, and efficiency. The current scope and limitation of the work are that it is best suited for missions that's duration is min 10mins to a maximum of 3 h. In the future, we are working on making the protocol adaptive and self-deciding to apply different levels of encryption based on the mission requirements.

Acknowledgement: The authors acknowledge the Center for Smart Society 5.0 [CSS5], School of Computing and Engineering, Taylor's University, for their support to complete this research.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] Y. Zeng, R. Zhang and T. J. Lim, "Wireless communications with unmanned aerial vehicles: Opportunities and challenges," *IEEE Communications Magazine*, vol. 54, no. 5, pp. 36–42, 2016.

[2] J. Janousek and P. Marcon, "Precision landing options in unmanned aerial vehicles," in *2018 Int. Interdisciplinary PhD Workshop*, Poland, pp. 58–60, 2018.

[3] N. A. Khan, S. N. Brohi and N. Z. Jhanjhi, "UAV's applications, architecture, security issues and attack scenarios: A survey," in *Intelligent Computing and Innovation on Data Science*. Berlin, Germany: Springer, pp. 753–760, 2020.

[4] M. H. Choi, B. Shirinzadeh and R. Porter, "System identification-based sliding mode control for small-scaled autonomous aerial vehicles with unknown aerodynamics derivatives," *IIEEE/ASME Transactions on Mechatronics*, vol. 21, no. 6, pp. 2944–2952, 2016.

[5] B. G. Maciel-Pearson, S. Akçay, A. Atapour-Abarghouei, C. Holder and T. P. Breckon, "Multi-task regression-based learning for autonomous unmanned aerial vehicle flight control within unstructured outdoor environments," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4116–4123, 2019.

[6] W. Kwon, J. H. Park, M. Lee, J. Her, S.-H. Kim *et al.,* "Robust autonomous navigation of unmanned aerial vehicles (UAVs) for warehouses' inventory application," *IEEE Robotics and Automation Letters*, vol. 5, no. 1, pp. 243–249, 2019.

[7] T. Samad, J. S. Bay and D. Godbole, "Network-centric systems for military operations in urban terrain: The role of UAVs," *Proc. of the IEEE*, vol. 95, no. 1, pp. 92–107, 2007.

[8] V. Roberge, M. Tarbouchi and G. Labonté, "Fast genetic algorithm path planner for fixed-wing military UAV using GPU," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 54, no. 5, pp. 2105–2117, 2018.

[9] D. Orfanus, E. P. de Freitas and F. Eliassen, "Self-organization as a supporting paradigm for military UAV relay networks," *IEEE Communications Letters*, vol. 20, no. 4, pp. 804–807, 2016.

[10] M. Saleh, N. Jhanjhi and A. Abdullah, "Proposing a privacy protection model in case of civilian drone," in *Proc. 2020 22nd Int. Conf. on Advanced Communication Technology*, Phoenix Park, South Korea, pp. 596–602, 2020.

[11] G. Quiroz and S. J. Kim, "A confetti drone: Exploring drone entertainment," in *Proc. 2017 IEEE Int. Conf. on Consumer Electronics*, Las Vegas, USA, pp. 378–381, 2017.

[12] D. Murugan, A. Garg and D. Singh, "Development of an adaptive approach for precision agriculture monitoring with drone and satellite data," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 12, pp. 5322–5328, 2017.

[13] A. M. Jawad, H. M. Jawad, R. Nordin, S. K. Gharghan, N. F. Abdullah *et al.,* "Wireless power transfer with magnetic resonator coupling and sleep/active strategy for a drone charging station in smart agriculture," *IEEE Access*, vol. 7, no. 1, pp. 139839–139851, 2019.

[14] I. Bor-Yaliniz, S. S. Szyszkowicz and H. Yanikomeroglu, "Environment-aware drone-base-station placements in modern metropolitans," *IEEE Wireless Communications Letters*, vol. 7, no. 3, pp. 372–375, 2017.

[15] A. N. Chaves, P. S. Cugnasca and J. Jose, "Adaptive search control applied to search and rescue operations using unmanned aerial vehicles (UAVs)," *IEEE Latin America Transactions*, vol. 12, no. 7, pp. 1278–1283, 2014.

[16] N. A. Khan, N. Z. Jhanjhi, S. N. Brohi, R. S. A. Usmani and A. Nayyar, "Smart traffic monitoring system using unmanned aerial vehicles (UAVs)," *Computer Communications*, vol. 157, no. 1, pp. 434–443, 2020.

[17] D. Wang, P. Hu, J. Du, P. Zhou, T. Deng *et al.,* "Routing and scheduling for hybrid truck-drone collaborative parcel delivery with independent and truck-carried drones," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10483–10495, 2019.

[18] A. Tariq, S. M. Osama and A. Gillani, "Development of a low cost and light weight UAV for photogrammetry and precision land mapping using aerial imagery," in *Proc. 2016 Int. Conf. on Frontiers of Information Technology*, Islamabad, Pakistan, pp. 360–364, 2016.

[19] A. Gurtner, D. G. Greer, R. Glassock, L. Mejias, R. A. Walker *et al.,* "Investigation of fish-eye lenses for small-UAV aerial photography," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 3, pp. 709–721, 2009.

[20] I. Mademlis, V. Mygdalis, N. Nikolaidis, M. Montagnuolo, F. Negro *et al.,* "High-level multiple-UAV cinematography tools for covering outdoor events," *IEEE Transactions on Broadcasting*, vol. 65, no. 3, pp. 627–635, 2019.

[21] E. Natalizio, N. R. Zema, E. Yanmaz, L. D. P. Pugliese and F. Guerriero, "Take the field from your smartphone: Leveraging UAVs for event filming," *IEEE Transactions on Mobile Computing*, vol. 19, no. 8, pp. 1971–1983, 2019.

[22] G. D. Georgiev, G. Hristov, P. Zahariev and D. Kinaneva, "Forest monitoring system for early fire detection based on convolutional neural network and UAV imagery," in *Proc. 2020 28th National Conf. with International Participation*, Sofia, Bulgaria, pp. 57–60, 2020.

[23] C. Chen, C. Grier, A. Malfa, M. Booen, C. Xia *et al.,* "High-speed optical links for UAV applications," in *Free-Space Laser Communication and Atmospheric Propagation XXIX*. vol. 10096. California, United States, 1009615–1009626, 2017.

[24] J. Russell, "Facebook is reportedly testing solar-powered internet drones again—this time with Airbus, techcrunch.com," 2019. [Online]. Available: https://techcrunch.com/2019/01/21/facebook-airbus-solar-drones-internet-program/ (Accessed Feb. 28, 2021).

[25] E. Salamí, C. Barrado and E. Pastor, "UAV flight experiments applied to the remote sensing of vegetated areas," *Remote Sensing*, vol. 6, no. 11, pp. 11051–11081, 2014.

[26] S. Berrahal, J.-H. Kim, S. Rekhis, N. Boudriga, D. Wilkins *et al.,* "Border surveillance monitoring using quadcopter UAV-aided wireless sensor networks," *Journal of Communications Software and Systems*, vol. 12, no. 1, pp. 67–82, 2016.

[27] L. Krichen, M. Fourati and L. C. Fourati, "Communication architecture for unmanned aerial vehicle system," in *Int. Conf. on Ad-Hoc Networks and Wireless*, pp. 213–225, 1st ed., Chap. 1, Sec. 1. USA, Cham, Springer, 2018.

[28] J. Gertler, "US unmanned aerial systems," *Library of Congress Washington DC Congressional Research Service*, vol. 1, no. 1, pp. 1–10, 2012.

[29] N. A. Khan, N. Jhanjhi, S. N. Brohi and A. Nayyar, "Emerging use of UAV's: Secure communication protocol issues and challenges," in *Drones in Smart-cities: Security and Performance*, 1st ed., vol. 1. pp. 37–55, Chap. 3, Sec. 1. Turkey: Elsevier, 2020.

[30] U. development Team, "UAVCAN Communication Protocol," 2014. [Online]. Available: https://uavcan.org/Specification/1._Introduction/ (Accessed Aug. 28, 2019).

[31] V. Kriz and P. Gabrlik, "Uranuslink-communication protocol for uav with small overhead and encryption ability," *IFAC-PapersOnLine*, vol. 48, no. 4, pp. 474–479, 2015.

[32] Y.-M. Kwon, "Vulnerability analysis of the Mavlink protocol for Unmanned Aerial Vehicles," Ph.D. dissertation. Univerity of Daegu Gyeongbuk Institute of Science and Technology (DGIST), 2018.

[33] S. Atoev, K.-R. Kwon, S.-H. Lee and K.-S. Moon, "Data analysis of the MAVLink communication protocol," in *Proc. 2017 Int. Conf. on Information Science and Communications Technologies*, Tashkent, Uzbekistan, pp. 1–3, 2017.

[34] S. Veena, S. Vaitheeswaran and H. Lokesha, "Towards the development of secure mavs," in *Proc. ICRAMAV-2014 (3rd Int. Conf.)*, India, pp. 1–10, 2014.

[35] A. Koubâa and B. Qureshi, "Dronetrack: Cloud-based real-time object tracking using unmanned aerial vehicles over the internet," *IEEE Access*, vol. 6, no. 1, pp. 13810–13824, 2018.

[36] AD Team, "Ardupilot project," 2009. [Online]. Available: https://ardupilot.org/index.php/about (Accessed Feb. 28, 2021).
[37] A. Allouch, O. Cheikhrouhou, A. Koubaa, M. Khalgui and T. Abbes, "MAVSec: Securing the MAVLink protocol for ardupilot/PX4 unmanned aerial systems," in *Proc. 2019 15th Int. Wireless Communications & Mobile Computing Conf.*, Tangier, Morocco, pp. 1–9, 2019.
[38] PD Team, "PX4 Autopilot," 2012. [Online]. Available: https://px4.io/ (Accessed Feb. 28, 2021).
[39] A. Tridgell and L. Meier, "MAVLink 2.0 packet signing proposal," Scientific report, 2015. [Online]. Available: https://lists.linuxfoundation.org/pipermail/dronecode-tsc/2015-October/000171.html.
[40] G. William, B. Elizabeth, B. Mike, S. Daniel, S. Ryan *et al.,* "Challenges of securing and defending unmanned aerial vehicles," in *National Cyber Summit (NCS) Research Track*, 1st ed., vol. 1. New York, USA: Springer, pp. 119–138, 2020.
[41] N. Hoque, D. K. Bhattacharyya and J. K. Kalita, "Botnet in DDoS attacks: Trends and challenges," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2242–2270, 2015.
[42] G. Vasconcelos, G. Carrijo, R. Miani, J. Souza and V. Guizilini, "The impact of DoS attacks on the AR. Drone 2.0," in *Proc. 2016 XIII Latin American Robotics Symp. and IV Brazilian Robotics Symp.*, Recife, Brazil, pp. 127–132, 2016.
[43] J. Whelan, A. Almehmadi, J. Braverman and K. El-Khatib, "Threat Analysis of a long range autonomous unmanned aerial system," in *Proc. 2020 Int. Conf. on Computing and Information Technology*, Tabuk, Saudi Arabia, pp. 1–5, 2020.
[44] K. Yoon, D. Park, Y. Yim, K. Kim, S. K. Yang *et al.,* "Security authentication system using encrypted channel on UAV network," in *Proc. 2017 First IEEE Int. Conf. on Robotic Computing*, Taichung, Taiwan, pp. 393–398, 2017.
[45] A. Shoufan, H. AlNoon and J. Baek, "Secure communication in civil drones," in *Int. Conf. on Information Systems Security and Privacy*, cham, Springer, pp. 177–195, 2015.
[46] B. S. Rajatha, C. M. Ananda and S. Nagaraj, "Authentication of MAV communication using Caesar Cipher cryptography," in *2015 Int. Conf. on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials*, Avadi, India, pp. 58–63, 2015.
[47] J. A. Marty, "Vulnerability analysis of the mavlink protocol for command and control of unmanned aircraft," MS dessertation, Air Force Institute of Technology, 2013.
[48] R. Hamsavahini, S. Varun and S. Narayana, "Development of light weight algorithms in a customized communication protocol for micro air vehicles," *International Journal Of Latest Research In Science And Technology*, vol. 1, no. 1, pp. 73–79, 2016.
[49] N. Butcher, A. Stewart and S. Biaz, Securing the mavlink communication protocol for unmanned aircraft systems. in *Technical Report # CSSE 1402 (2014)*. USA: Appalach State University. Auburn University, 2013.
[50] J. Bian, R. Seker and M. Xie, "A secure communication framework for large-scale unmanned aircraft systems," in *Proc. 2013 Integrated Communications, Navigation and Surveillance Conf.*, Herndon, VA, USA, pp. 1–12, 2013.
[51] R. Altawy and A. M. Youssef, "Security, privacy, and safety aspects of civilian drones: A survey," *ACM Transactions on Cyber-Physical Systems*, vol. 1, no. 2, pp. 7, 2017.
[52] A. Koubaa, A. Allouch, M. Alajlan, Y. Javed, A. Belghith *et al.,* "Micro air vehicle link (MAVLink) in a nutshell: A Survey," *IEEE Access*, vol. 7, no. 1, pp. 87658– 87680, 2019.
[53] F. T. AL-Dhief, N. Sabir, N. M. A.Latif, N. N. N. A. Malik, M. A. A. Albader *et al.,* "Performance comparison between Tcp And Udp protocols in different simulation scenarios," *International Journal of Engineering & Technology*, vol. 7, no. 4.36, pp. 172–176, 2018.
[54] W. Goralski, "The illustrated network: How TCP/IP works in a modern network," in *The Illustrated Network: How TCP/IP Works in a Modern Network Morgan Kaufmann*, 2 ed., Chennai, India: Elsevier, 2017.
[55] Anis Koubaa, "MAVSec: Securing the MAVLink protocol for ardupilot and PX4 unmanned aerial systems," 2019. [Online]. Available: https://www.slideshare.net/AnisKoubaa/mavsec-securing-the-mavlink-protocol-for-ardupilot-and-px4-unmanned-aerial-systems (Accessed Mar. 17, 2021).