DEU FMD 26(77) (2024) 350-355

Dokuz Eylül Üniversitesi Mühendislik Fakültesi Fen ve Mühendislik Dergisi
Dokuz Eylul University Faculty of Engineering Journal of Science and Engineering
Elektronik/Online ISSN: 2547-958X

RESEARCH ARTICLE / ARAŞTIRMA MAKALESI

Secure Communication Between Unmanned Aerial Vehicle and Ground Control Station

İnsansız Hava Aracı ve Yer Kontrol İstasyonu Arasında Güvenli İletişim

Pınar Savaştürk [1], İbrahim Atakan Kubilay [*2], Gökhan Dalkılıç [3]

[1] Kırklareli Üniversitesi, Mühendislik Fakültesi Yazılım Mühendisliği Bölümü, Kırklareli, TÜRKİYE
[2,3] Dokuz Eylül Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü, İzmir, TÜRKİYE
Corresponding Author / Sorumlu Yazar *: atakan.kubilay@deu.edu.tr

Abstract

The main purpose of this article is to provide data transfer by providing secure communication with drones, which is a kind of unmanned aerial vehicle (UAV). In our study, without intervening hardware and software of the drone (most of the drones do not allow interference with their hardware and software), a security layer is added for data communication. In our study for adding this layer, Wemos D1 Mini, having wireless network module and cheap, was used. In the scope of the study, Wi-Fi modules are customized and converted into Access Point and Client forms. The message sent to the specialized modules is provided through the Mobile Application, which has been designed and implemented for this study. Thus, commands were transmitted in an encrypted form in the wireless data transmission environment. As a result, this concept has become a prototype that can be customized according to any specific drone having Wi-Fi communication.

Keywords: *UAV secure communication, IoT secure communication, lightweight encryption*

Öz

Bu makalenin temel amacı, bir tür insansız hava aracı (İHA) olan dronlarla (uçangöz) güvenli iletişim sağlayarak veri aktarımı sağlamaktır. Çalışmamızda, dronun donanım ve yazılımına müdahale etmeden (birçok dron donanım ve yazılımına müdahele edilmesine izin vermez) veri aktarımında bir güvenlik katmanı eklenmiştir. Çalışmamızda bu katmanı eklemek için kablosuz ağ modülüne sahip ve maliyeti düşük olan Wemos D1 Mini kullanılmıştır. Çalışma kapsamında Wi-Fi modülleri özelleştirilmiş ve Erişim Noktası ve İstemci formlarına dönüştürülmüştür. Özelleşmiş modüllerine gönderilen mesaj, bu çalışma için tasarlanan ve geliştirilen Mobil Uygulama aracılığıyla sağlanmıştır. Böylece, komutlar kablosuz veri iletim ortamında şifrelenmiş bir biçimde iletilmiştir. Sonuç olarak, bu konsept, Wi-Fi iletişime sahip her türlü dron için özelleştirilebilen bir prototip haline gelmiştir.

Anahtar Kelimeler: *İHA güvenli iletişim, IoT güvenli iletişim, düşük seviyeli şifreleme*

1. Introduction

Security is essential for all structures. In particular, devices that communicate wirelessly need further protection. Basically, extra measures should be taken to ensure the control and safety of such devices. Ensuring both safety and control is very important. The biggest problem in achieving this control is that the transmitter and receiver cannot be accurately verified due to wireless transmission [1]. Especially drones that communicate wirelessly are exposed to security problems. The drone, which is controlled by any user, can be attacked, seized, or damaged. Hacking of drones is an important challenge, and a list of events outlined in [2] show that commandeering of commercial drones is quite common, and even military drones can be victim to outside attacks. Therefore, network security is a vital concept for safe drone operations.

Article [3] describes a Man-in-the-Middle (MitM) attack that can be used to control a drone from kilometers away. The same source has proposed various encryption schemes at chip and application levels as possible solutions for the problem, please see [4] for a more general review of network security problems of drones.

The main reason for the problem is that the data transmitted between the control station and the drone cannot be properly protected. In particular, the data transmitted clearly between this structure causes many problems. Because of these situations, there is a need for improvements in this area in order to ensure secure communication and to protect the transmitted data. In our study, the connection between the control station and the unmanned aerial vehicle (UAV) [5] was secured with additional hardware components such as the Wemos D1 Mini wireless fidelity (Wi-Fi) module. This study focuses on the UAV like DJI Tello Drone.

2. Related Works

The project area covers drones, a sub-class of unmanned aerial vehicles [6]. It is necessary to ensure the safe communication of these unmanned aerial vehicles. In this context [7] is a good guide. As mentioned in the article, unmanned aerial vehicles can be used in many areas and they may be exposed to many threats [8]. These attacks can sometimes take control of an unmanned aircraft, render it dysfunctional or disable and damage it. Under the control of malicious users, these devices can become very dangerous. It can even threaten a person's life. In this article, it is

DOI:10.21205/deufmd.2024267720

Geliş Tarihi / Received: 17.03.2023
Kabul Tarihi / Accepted: 08.08.2023

Atıf şekli / How to cite:
Savasturk, P., Kubilay , I.A., Dalkilic, G. (2024) Secure Communication Between Unmanned Aerial Vehicle and Ground Control Station. DEUFMD, 26(77), 350-355.

stated that uses of unmanned aerial vehicles range from education to industry, from trade to defense [9]. It explains the deficits of unmanned aerial vehicles that communicate with wireless communication channels [6]. Radio waves, electromagnetic waves, and the communication channels they use to communicate are explained in the article. From this point of view, they have made improvements on how the wireless communication channels can be more secure [10]. In the article, attention has been paid to the communication network between the sender and the receiver. The most important points of how unmanned aerial vehicles are used, how much an attack can cause damage and developing possible solutions to the attacks are discussed.

As seen in article [11], attacks can be made to secure the drones. Models were made for attack types. Thus, possible threats were identified. The basis of the modeling is a controller that controls the drones with any application or device. The controller is connected to the drone via a wireless network. When the attacker who intercepts the connection intercepts the drone, the drone is out of control according to the legitimate user. Thus, the attacker has all the information and control of the drone. The major contribution of this article is to realize the gap between the controller and the drone. In all of the modeled attacks, the user is directly connected to the drone. Therefore, the attacker is trying to connect to the drone to capture the drone. Due to insufficient encryption in wireless networks [12], these attacks are often successful. For such attacks, a solution has been produced in our study.

In article [13], there is a drone in the given architecture. There is a control console that controls the drone. In general, the man in the middle attack between the controller console and the drone is described. A mechanism called xBee is placed in the ancestor to improve the mechanism that is likely to be exposed to this attack. In this way, the communication channel expected to be provided between the drone and the controller becomes gradual. In the architecture we designed, we provide 2 wireless network modules together. However, the module inserted in the discussed article [13] is only one. And the connection to it is made wirelessly. The architecture in the discussed article does not mention the cryptography structure to be used in the transmission of data. Advanced encryption standard (AES) encryption was used in the communication channel in our study. The use of number generators that would allow stream encryption on the Wemos D1 Mini is not useful. In addition, AES block cipher using strong, s-box and permutation embedded on the module is preferred. Confusion and diffusion also favored the use of block ciphers. At the same time, thanks to the CFB (Cipher FeedBack) mode used by AES, it is possible to produce results such as stream ciphers. A high level of security is provided with a 256-bit long key. In this article, the density that can be experienced in data transmission is mentioned. With this point, we provide data transfer with user datagram protocol (UDP) in order to control the data transfer in our study. Another important point in this regard is the protocol on which the commands to be sent from the controller to the drone. Since the devices used are Internet of things (IoT) devices, there are no high-capacity processors or memory. Therefore, the architecture should be designed taking these points into account. Since the devices we are working on are flying devices, we opted for smaller, less powerful devices to minimize weight. In other words, the main purpose is to provide maximum efficiency with our hardware assets. The most important contribution we extracted from the whole study was the idea of placing a buffer mechanism between the communication channel to improve security. This is one of the fundamental points for our study. Another feature of our article is that the architecture described is turned into a prototype.

In the article [14], the modeling of the quadrotor DJI Tello drone in Python using fuzzy logic in artificial neural networks is presented. This modeling has also been implemented by Python. By controlling the drones, which are expressed in the study, starting to take up a large space in technologies; human detection/follow-up, analysis. Detection of data that does not have the class attribute used in classical machine learning models (for example, images flowing as streams in the drone) has been demonstrated.

In the study [15], the image taken from the camera autonomously and the determination of the location of the drone according to its own coordinates are presented. Due to its physical structure, the low cost of the algorithms running on the drone directly affects the flight time/power from the parallel measure. For this reason, in this study, object detection developed on the robot operating system based on MATLAB and a system that can detect the path by avoiding obstacles has been designed and implemented. At the same time, the results are clearly understood as the results are simulated with virtual reality.

Study [16] examines the situation of taking control of drones with cyber-attacks and focuses on the control in the wireless channel. Attack scenarios were analyzed, and defense situations were put forward and implemented. In the wireless network, it has been focused on not only passive protection with password settings but also active protection against injections.

3. Materials and Methods

According to our study, an extra two Wi-Fi modules are needed to establish the architecture of secure communication. The main purpose of our study is to transfer the commands controlling drone, which is unmanned aerial vehicle, in a secure environment. Therefore, the modules used satisfy different functionalities.

There are two modules (ESP32 (1) and ESP (2) from Figure 1) interconnected by a physical cable. These modules are attached to the drone. One of the modules (ESP32 (2)) was turned into an access point. This means that when the module is connected to the power unit, an access point called service set identifier (SSID) that we define becomes visible on the Wi-Fi network. We have a controller person that controls the drone using the base station. In this study, the controller uses a mobile application to control the drone. The controller first connects [17] to the access point (Figure 1 ESP (2)). (S)he then enters the application on his/her smartphone, and s(he) encounters an interface with commands to control the drone.

The task of the module (ESP (2)) is to take encrypted commands from the controller via Wi-Fi and after decrypting the commands to transmit those to the other module (ESP (1)) via cable. The task of the next module (ESP (1)) is to take commands from the first module (ESP (2)) via cable and to transmit those commands to the drone via Wi-Fi using the special port and user datagram protocol (UDP). One reason to transfer the commands with UDP is because it's known that the drone allows it as written in the drone's software development kit (SDK) document. This module can also be called the client because it provides connection to the drone. The drone also has an access point feature.

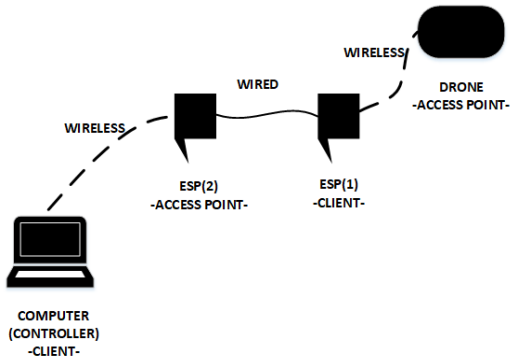

Figure 1. Diagram of the proposed general mechanism.

One of the basic building blocks of the security in this mechanism is the mobile application. Because the mobile application encrypts the relevant command selected from the interface with AES and transmits it to the Wi-Fi module (ESP (2)), which is an access point, wirelessly. This module (ESP (2)) decrypts the command and transfers it to the other module via cable. The module that receives the command with the decrypted code forwards it to the drone and the drone executes the command. The main purpose of using buffer modules is to perform encryption operations and add a security layer to any kind of drone having a Wi-Fi connection.

In Figure 1, the described general mechanism diagram is schematized. There are two access points in this structure. One of them is our customized Wi-Fi module and the other is the drone. If no intervention is made, both the access point and the drone we defined will appear on the controller's Wi-Fi network. Therefore, there may be a dilemma. In our study, we designed a structure to prevent this dilemma. We covered the main frame of the drone with aluminum foil and steel fabric and put the module that sends the commands to the drone wirelessly with UDP in this structure which was covered. In this way, the drone with its 2.4 GHz broadcasting antenna has been covered and its broadcasting power has been reduced. The other module was placed on the drone with the cable extension. Since this module is an access point, it should be visible on the public network.

3.1. Encryption and decryption mechanism

In our study, the encryption processes of the command to be sent are prepared on the mobile application. Encryption operations were carried out with AES and Base64 structures. While developing applications in Android environment, java/security, javax/crypto and libraries in which some functions were made special were used. The special here is to change the attribute formats that the functions take in accordance with the format of the incoming data. The purpose of this customization is to enable the encryption to be decrypted on an IoT device, the Wemos D1 Mini. Decryption is proceeded on the Wi-Fi module in the form of an access point (Figure 1, ESP (2)).

In our study, the Android application acts as the UDP Server. The Wi-Fi module in the Access Point form is also defined as the UDP Client. After encryption is proceeded on the application, the message is sent to the UDP Client with the internet protocol (IP) address of the module (192.168.4.1) and the specially defined port number (6868). It is preferred over TLS as it has UDP applicability with DJI Tello Drone. Although TLS is used in other model drones, UDP has been preferred for Tello in terms of both its compatibility with the encryption structure and its usefulness. The Wi-Fi module (Figure 1, ESP (2)) reads incoming messages from the UDP Server in packets of 512 bytes. Receiving the message as encrypted, UDP Client decrypts the message it receives in accordance with its encryption. In this way, data flows in the wireless network encrypted. Commands that are decrypted in the module are sent to the other module in plaintext with the help of a small cable. While doing this sending process, the transmitter (TX) pin of the sending module is connected with the receiver (RX) pin of the receiving module. The sender writes the message (s)he will send to the serial platform. The recipient can also get the message by reading from the same serial platform.

This configuration prevents a man-in-the-middle attack, because the wireless connection between the computer and the drone is encrypted. The ESP modules are on the drone, connected by a wire, so unless an attacker can physically access the drone, there is no way for them to intrude the unencrypted communication. The second module relays the commands to the drone in unencrypted form, but because it's very close to the drone, it can't be affected by an attacker away from the drone.

In Figure 2 architectural view, the general mechanism of the study is summarized. The controller (ground control station) is connected wirelessly to the Wemos D1 Mini, which is an Access Point. The controller sends the encrypted command to the Access Point. The Access Point decrypts the command it receives in encrypted form. It then sends the command to the Wemos D1 Mini in Client mode to which it is wired. Client receiving the decoded command connects wirelessly to the drone and sends the command. The drone executes the command and sends the result of the execution to the Client. Wemos D1 Mini, which is one of the two wireless network devices used, makes the communication between the controller and the drone secure. The Client and Access Point actually function as if they were a single piece of hardware.

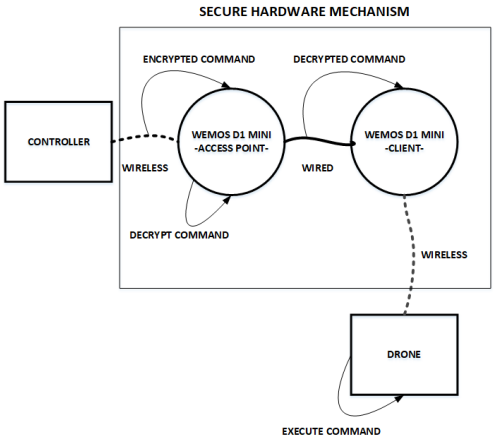

Figure 2. Architectural view.

3.2. Executing commands on drone

Wemos D1 Mini module, which receives commands with cable by decrypting, acts as a kind of client. Because this module connects to Tello DJI Drone that has the feature of Access Point automatically. It is this module that processes commands on the drone. This module communicates with the drone using the contact information specified in the SDK of the DJI Tello Drone.

The module, which establishes a wireless connection with the drone to operate the commands, communicates with the UDP. In this way, the operation requests from the user are made possible to operate the drone. It also returns a message when the drone is running or unable to run the messages it receives. In this message, it transmits the data to the module with UDP.

3.3. Implementation platforms

While performing this work, one Wemos D1 Mini, two separate Wi-Fi modules, and one mobile application were programmed.

Arduino integrated development environment (IDE) was used for programming the Wi-Fi modules [18]. We also used Android Studio while developing the mobile application.

The Android application was implemented on the Java platform. This application is defined as the UDP Server. The address and port to communicate as code forced were determined. After the user enters the command he wants to run on the screen, the command is encrypted using the relevant encryption structure and it is AES, Base64 in our study. Using the encrypted message and contact information, the message is sent to the Access Point in the form of UDP Client via UDP connection. For the Access Point to receive the sent encrypted message, the mobile application must be connected to the module that is the Access Point on the device's Wi-Fi network [19]. The application uses the UDP Socket connection and datagram packets when sending messages. Whenever a command is sent, it is encrypted to a distinct ciphertext, so that eavesdropper cannot make a guess which encrypted data corresponds to which command by using previously collected data.

While programming the modules acting as Access Points, we used libraries such as ESP8266WiFi, WifiUdp, AES, Base64. When we made the device visible in wireless networks, we assigned SSID and password. The Access Point then waits for the connection of the mobile application. Since the mobile application acts as UDP Server, the module opens the 6868 port where it will communicate with the server after the connection is established. Operations are initialized in the "setup" function. The "loop" function listens for requests from the UDP Server so that it can be answered. Until reaching the size of the package defined jointly between UDP Server and Client, the package is listened. The incoming packet is then decrypted. The decrypted message is written to the serial platform that it is wired to the other module.

Arduino IDE was used for programming the EPS32 modules. Wire, ESP8266Wifi and WifiUdp libraries were used while coding these modules. The Android application first tries to connect with the drone in the "setup" function. The "setup" function is used to define the initial values required for the program. For example: defining the port, specifying the constants to use, etc. After connection is established, UDP communication is initiated. In order to operate the commands on the DJI Tello drone, it is necessary to send the command "command" first. After the UDP connection is provided, this special start command is sent. Then the data coming from the serial platform is read inside the loop function that runs continuously. Thanks to the "loop" function, incoming commands can be handled continuously. When the related commands arrive to the EPS32 module (Figure 1, ESP (1)), it sends them to the drone. In this way, control of the drone is provided.

4. Results

In this project, improvements (The structure shown in Figure 2) were made on drone commands to be safely operated and transferred. According to these improvements, some tests were carried out.

According to the results obtained from this study, received encrypted messages are given in Table 1.

It is seen that the messages received from the controller are encrypted and when the same message is sent again, it sends unique encrypted messages. The "mixed" command shown in this table is not available in the SDK. This command was created by combining "take off, flip left and land" commands. The messages shown in the table are obtained from the Access Point (Figure 1, ESP (2)). Encrypted messages sent from the Mobile Application, which is the controller, are obtained from the Access Point. In addition, this table shows us that the command transmitted on the Wi-Fi network that provides wireless communication is definitely not in understandable and readable form. This plays an important role in the safe transmission of commands transmitted for drone control. This added security layer contributes to the security part, which is the most critical point of unmanned aerial vehicles. In this way, an important improvement has been achieved.

Table 1. Received encrypted message.

Command Received	Encrypted Message
Mixed	lYyvabuBVdKmC0uHPcrp3g==
Mixed	nKhs3tUrtKnkScXIMcKZsQ==
Mixed	pOmSJeLXl2p7RcHQfwCGjQ==
Land	JsiuSDCpKgetvNuVlIPj8g==
Land	AvX3ig0V3c/yVaMhaN2yqQ==
Flip Left	77o6CNCINsj2kJjSqVmqxQ==
Flip Left	IracWH2ZeYIlALJv59RePw==
Takeoff	Z2S5m3P95xHKhH9I3QqIPg==
Takeoff	995QLzaoWuTOF/Zk9UrqRw==

Encrypted message transmission durations (Table 2) are prepared with the results obtained from the test studies. These durations are from the Controller to the Access Point. The end time is the time before transferring to the Wemos D1 Mini, which has a client feature, after decoding at the Access Point. These periods were obtained from the time outputs were given to the serial monitor of Arduino IDE. The stability of the system was checked by sending the same commands one after another for testing purposes. As shown in the table, regardless of the command sent, it can usually be transmitted around in 3 milliseconds. In addition, this table shows us that the command sent in encrypted does not cause a significant delay to slow down the operation. This is also very important for drones. For message transmission durations, at least 10 samples were taken for each command and their averages were taken.

Table 2. Encrypted message transmission durations.

Encrypted Received Message	Encrypted Transmission Duration (ms)
Mixed	2.6
Take off	3.5
Flip left	3
Land	3

The tests performed in calculating the values shown in Table 3 Plaintext Message Transmission Durations were used. While performing the calculations in this table, it is assumed that the Controller sends a command without encryption, explicitly sending the plaintext commands. This means that the data is sent without encryption in wireless network. When measuring the processes, the start time is that the Access Point receives the command, and the end time is sent to the Wemos D1 Mini with a cable connection. As can be seen from Table 3, most of these values are 0. This means that the added wired connection reduces the transmission time. This shows the usefulness of the mechanism designed in this study and the consistency of the

solution. While obtaining the data in Table 2 Encrypted Message Transmission Durations and Table 3 Plaintext Message Transmission Durations, at least 10 samples were taken for each command and their averages were taken.

Table 3. Plaintext message transmission durations.

Plaintext Received Message	Plaintext Transmission Duration (ms)
Mixed	0.33
Take off	0.5
Flip left	0
Land	0

The results of the study and the results obtained without using any security architecture are given in Table 4. Execution times are given for each command specified according to this table.

Table 4. With and without secure architecture command execution time comparison.

Command	Without Secure Architecture (ms)	With Secure Architecture (ms)
Take off	6651	7512
Flip left	2870	2462
Land	3284	4152

The data received in the Without Secure Architecture column were obtained as a result of direct communication between the computer and the drone. While obtaining these calculations, a software program developed with Node.js was run. During these tests, 10 different samples were collected for each command. For each sample, the start time was when the command was sent, and the end time was when the command was run in the drone. The averages of the received values were taken and given in the table.

The results in the With Secure Architecture column were obtained as a result of running commands over the Security Architecture described in the article. While calculating these processes, the start time is to send the command with an encrypted mobile application and the end time is to execute the command in the drone. These values were found by averaging at least 10 samples for each command.

This table was prepared to compare the security mechanism developed in the execution of commands in general with the system that the drone is directly linked to. The results show that with the hardware part put to ensure the security of the structure, the encryption and decryption mechanisms did not create too many differences that could create control problems in practice. In fact, the values are too small to notice the time difference for humans.

5. Discussion and Conclusion

In this study, Wemos D1 Minis with two Wi-Fi modules were placed between the drone and the control station. One of these modules was converted to an Access Point and the other to a Client form. The module in the Client form was designed to provide a UDP connection with the drone. The Mobile Application, which acts as the controller, has been developed and the connection of the device with this application to the Wi-Fi module in Access Point form has been provided. Modules in the form of Client and Access Point were connected with a cable to provide serial communication. With this structure, encrypted transmission of commands sent from Mobile Application was provided. With the help of Wi-Fi modules put together, these encrypted messages are decrypted, and the drone is operated. Thanks to this structure, security was provided between the drone and the controller without creating a significant delay.

Ethics committee approval and conflict of interest statement

There is no need for an ethics committee approval in the current article.

There is no conflict of interest with any person/institution in the current article.

Author Contribution Statement

All authors contributed equally to this manuscript, and they have accepted responsibility for the entire content of this manuscript and approved its submission.

References

[1] Urien, P. 2018. An Innovative Four-Quarter IoT Secure Architecture Based on Secure Element. 14th Int. Wireless Communications & Mobile Computing Conf. (IWCMC), 25-29 June, Limassol, 1074-1080, DOI: 10.1109/IWCMC.2018.8450435.

[8] Kubilay, İ. A. and Kubilay, H. 2018. Drone Design for Abiding Legal Guidelines. International Conference on Science and Technology, 5-9 September, Prizren, 482-488.

[9] Rodday, N. M., Schmidt, R. D. O. and Pras, A. 2016. Exploring security vulnerabilities of unmanned aerial vehicles. NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium, 25-29 April, Istanbul, 993-994, DOI: 10.1109/NOMS.2016.7502939.

[10] Verup, M. and Olin, M. 2016. Security models and exploitations in theory and practice for unmanned aerial vehicles. http://www2.compute.dtu.dk/pubdb/edoc/ imm7054.pdp (Accessed on: 07.03.2023).

[5] Hartmann, K. and Giles, K. 2016. UAV Exploitation: A New Domain for Cyber Power. 8th Int. Conf. on Cyber Conflict, 31 May-03 June, Tallinn, 205-221, DOI: 10.1109/CYCON.2016.7529436.

[6] Bian, J., Seker, R. and Xie, M. 2013. A secure communication framework for large-scale unmanned aircraft systems. 2013 Integrated Communications, Navigation and Surveillance Conference, 22-25 April, Herndon, 1-12, DOI: 10.1109/ICNSurv.2013.6548542.

[7] Bian, J., Seker, R., Ramaswamy, S. and Yilmazer, N. 2009. Container communities: Anti-tampering Wireless Sensor Network for global cargo security. 17th Mediterranean Conference on Control and Automation, 24-26 June, Thessaloniki, 464-468, DOI: 10.1109/MED.2009.5164585.

[8] Mitchell, R. and Chen, I. R. 2014. Adaptive Intrusion Detection of Malicious Unmanned Air Vehicles Using Behavior Rule Specifications, IEEE Transactions on Systems, Man, and Cybernetics: Systems, Vol. 44, no. 5, pp. 593-604, DOI: 10.1109/TSMC.2013.2265083.

[9] Gupta, L., Jain, R. and Vaszkun, G. 2016. Survey of Important Issues in UAV Communication Networks, IEEE Communications Surveys & Tutorials, Vol. 18, no. 2, pp. 1123-1152. DOI: 10.1109/COMST.2015.2495297.

[10] Deva Sarma, H. K. and Kar, A. 2006. Security Threats in Wireless Sensor Networks. 40th Annual 2006 Int. Carnahan Conf. on Security Technology, 16-19 October, Lexington, 243-251, DOI: 10.1109/CCST.2006.313457.

[11] Samid, G. 2016. Drone Targeted Cryptography, IACR Cyrptol. ePrint Arch., pp. 499-506.

[12] He, D., Chan, S. and Guizani, M. 2017. Drone-Assisted Public Safety Networks: The Security Aspect, IEEE Communications Magazine, Vol. 55, no. 8, pp. 218-223, DOI: 10.1109/MCOM.2017.1600799CM.

[13] Singh, M., Rajan, M. A., Shivraj, V. L. and Balamuralidhar, P. 2015. Secure MQTT for Internet of Things (iot). Fifth Int. Conf. on Communication Systems and Network Technologies, 4-6 April, Gwalior, 746–751.

[14] Giernacki, W., Rao, J., Sladic, S., Bondyra, A., Retinger, M. and Espinoza-Fraire, T. DJI Tello Quadrotor as a Platform for Research and Education in Mobile Robotics and Control Engineering. 2022 Int. Conf. on Unmanned Aircraft Systems (ICUAS), 21-24 June, Dubrovnik, 735-744, DOI: 10.1109/ICUAS54217.2022.9836168.

[15] Giernacki, W., Kozierski, P., Michalski, J., Retinger, M., Madonski, R. and Campoy, P. 2020. Bebop 2 Quadrotor as a Platform for Research and Education in Robotics and Control Engineering. 2020 Int. Conf. on Unmanned Aircraft Systems (ICUAS), 1-4 September, Athens, 1733-1741, DOI: 10.1109/ICUAS48674.2020.9213872.

[16] Mamchenko, M. V. 2021. Analysis of Control Channel Cybersecurity of the Consumer-Grade UAV by the Example of DJI Tello, Journal of Physics: Conference Series, Vol. 1864, DOI: 10.1088/1742-6596/1864/1/012127.

[17] Radu, D., Cretu, A., Avram, C., Astilean, A. and Parrein, B. 2018. Video Content Transmission in a Public Safety System Model based on Flying Ad-Hoc Networks. 2018 IEEE Int. Conf. on Automation, Quality and Testing, Robotics (AQTR), 24-26 May, Cluj-Napoca, 1-4, DOI: 10.1109/AQTR.2018.8402713.
[18] DJI Company 2023. Tello SDK 2.0. https://dl-cdn.ryzerobotics.com/downloads/Tello/Tello%20SDK%202.0%20User%20Guide.pdf (Accessed on: 07.03.2023).
[19] Valente, J., Cardenas, A. A. 2017. Understanding Security Threats in Consumer Drones Through the Lens of the Discovery Quadcopter Family. 2017 Workshop on Internet of Things Security and Privacy, 30 October-3 November, Dallas, 31-36, DOI: 10.1145/3139937.3139943.