Secure Video Streaming Implementation for Unmanned Air Vehicle (UAV) Data Link with Raspberry Pi 3 over https

Mahir DURSUN, İsmet ÇUHADAR

Electrical and Electronics Engineering, Gazi University, Ankara, Turkey
mdursun@gazi.edu.tr, ismetcuhadar@gmail.com
(Geliş/Received:20.04.2017; Kabul/Accepted:13.01.2018)
DOI: 10.17671/gazibtd.307294

***Abctract*—** With the widespread usage of the UAVs, the questions about their reliabilities have increased recently. Because they are generally used for reconnaissance, surveillance and intelligence purposes, it is very important that air vehicles are physically robust and that software and data transfer is reliable. The first thing to cover when it comes to UAV security and reliability is the data link subsystem. Although UAVs are produced by many companies, data link/communication security problem still continues. It is very important to send the right information to the right spot over long distances. In this paper, a representation environment created for working with secure data link and a new method for video streaming for UAVs developed in this environment are presented. In the representation environment, a data link system has been designed with Raspberry Pi 3, Picamera and https for video streaming and the method has been verified by experiments. The secure image transfer method was formed by adapting different existing methods (Motion-Jpeg, TCP/IP, TLS/SSL and user authentication/authorization system) to a different area as UAV system and using them together rather than developing a new encryption method. This method will be successfully used in UAV systems due to its low cost and reliability.

***Keywords*—** UAV, data link security, raspberry Pi 3, picamera, https, video streaming, motion-jpeg

İnsansız Hava Aracı (İHA) Veri Linki İçin Raspberry Pi 3 ile https Üzerinden Güvenli Video Transferi Uygulaması

***Özet*—** İHA'ların yaygınlaşması ile birlikte, güvenilirlikleri ile ilgili sorular son zamanlarda artmıştır. Genellikle keşif, gözetleme ve istihbarat amacıyla kullanıldıkları için hava araçlarının fiziksel olarak sağlam olması, yazılım ve veri transferinin güvenilir olması çok önemlidir. İHA güvenliği ve güvenilirliği söz konusu olunca ilk ele alınması gereken unsur veri link alt sistemi. İHA'lar birçok firma tarafından üretilmekle birlikte, veri linki/iletişim güvenliği problemi halen devam etmektedir. Uzun mesafelerde doğru bilgiye doğru noktaya göndermek çok önemlidir. Bu makalede, güvenli veri linki üzerinde çalışmak için oluşturulmuş bir temsil ortamı ve bu ortamda İHA sistemlerinde video transferi için geliştirilen yeni bir yöntem sunulmuştur. Temsili ortamda, Raspberry Pi 3, Picamera ve https ile bir veri bağlantı sistemi tasarlanmış ve yöntem deneyler ile doğrulanmıştır. Güvenli görüntü aktarma yönteminde, var olan yöntemler (Motion-Jpeg, TCP/IP, TLS/SSL ve kullanıcı kimlik doğrulama/yetkilendirme sistemi) farklı bir alan olan İHA sistemine uyarlanmış ve birlikte kullanmıştır. Bu yöntem düşük maliyet ve güvenilirlik nedeniyle İHA sistemlerinde başarıyla kullanılabilmektedir.

***Anahtar Kelimeler*—** İHA, very linki güveliği, raspberry Pi 3, picamera, https, video aktarımı, motion-jpeg

1. INTRODUCTION

The UAVs/drones low cost and easy obtainability have also increased the questions about their reliabilities. As the UAVs themselves can be used as an attack tool, also the attacks made to them have become a situation that is often encountered. Because of these reasons, it is very important that air vehicles are physically robust and that software and data transfer is reliable. In order to achieve this goal, first concern to consider and improve is to increase the reliability of data link usage in UAVs. In both the world and our country, although UAVs are produced by many companies, data link/communication security problem still continues. As well as transmitting data to long distances, it is very important to send the right information to the right spot. Furthermore, prevention of frequent external interventions in data transmission is one of the indispensable elements of UAV systems.

There are a number of important issues to have encountered in the search of literature on data links in UAV. For instance; Robert Erra, Vincent Guyot, Loica Avanthey, Antoine Gademer and Laurent Beaudoin have developed a concept called k-ary which aims to protect the software and sensitive information in the UAVs/drones in 2012. According to this concept, it is stated that by using Shamir's Secret Sharing Scheme and Neville-Akenen algorithm together, high level security is provided and taking mod compatible with designated keys not only makes easy to encrypt with limited CPU of UAV but also makes difficult to break the encryption [1].

In Özgür Koray Şahingöz's study (2013), it has been aimed to develop multi-stage dynamic key management system for ensuring the data link security; and claimed that the solution is manageable and has better performance than single key management system [2].

In his study, Nils Rodday have examined the security weaknesses in the UAVs: hacking, GPS spoofing methods, Wi-Fi, Bluetooth, ZigBee, XBee, KillerBee, radio waves, Brute Forces attacks over data links in 2015. In accordance with this attack he developed some attack and counter-attack scenarios, and offered three different alternatives: XBee on-board encryption, hardware encryption and application layer encryption. As a result; he suggested that any of the proposed countermeasures could be implemented but each brought better equipment and higher production costs over the UAV [3].

In 2016 Chen Xiao, Lifeng Waog, Mengjiao Zhu and Wemdong Wang, examining the application scenario and encryption trends in UAVs proposed a scenario-based method named Information Utility Value Oriented Encryption Optimization (IEO) [4].

In order to develop a secure video/image transfer method for UAVs, it is needed to have a representative environment in order to work on the data links [5]. The environment has been design for this reason which is shown in Figure 1 provided an ambit in that software development, implementations and experiments are carried on.

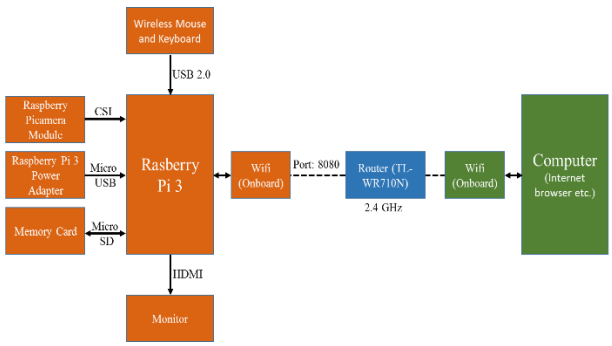

Figure 1. System general blog diagram

Raspberry Pi 3, a credit card size computer is used as first and most important piece. The other pieces of the environment are Raspberry Picamera Module (Picamera, Rev. 1.3), Raspberry Pi 3 power adapter, 16 GB Micro SD memory card, monitor, High Definition Multimedia Interface (HDMI) cable, wireless mouse and keyboard set and Router (TL-WR710N). The whole environment is given in Figure 2.

Figure 2. Representative environment

Raspberry Picamera Module (Rev. 1.3) has a camera with 5 megapixel (2592×1944) resolution and fixed-focus-point. It can provide 30 fps at 1080p, 60 fps at 720p or 60/90 at 640x480p video streaming.

The Router has been used to create a local area network (LAN) to provide the necessary connection between the computer and Raspberry Pi 3 to represent the data link between the aircraft and the ground data terminal with a frequency of 2.4 GHz.

A newly developed NOOBS operating system for beginner users has been installed on Raspberry Pi 3. The Go programming language has been preferred because of the simplicity and common libraries for the study [6].

After developing the software and making the experiments-applications successfully in the representative environment, the image transfer system was installed on a drone (quadcopter) as in Figure 3.

Figure 3. Image transfer system installed on drone

2. LIBRARY AND PROTOCOLS USED FOR IMAGE TRANSFER

For video streaming and transmission with Motion JPEG (mjpg); HTTP (Hyper Text Transfer Protocol), TCP/IP (Transmission Control Protocol/Internet Protocol) and "video for Linux" module have been used. The go-webcam library [7] by Oleksandr Senkovych has also been used. Motion-JPEG has been used because of its advantages;

- It's simple to implement because it uses a mature compression standard (JPG) in well-developed libraries.

- Mjpg can easily overcome rapid changes in the video stream, while the other methods (such as MPEG1, MPEG2 or H.264 / MPEG-4) can experience unacceptable quality loss when significantly changing frames.

- While the largest web browsers such as Safari, Google Chrome, Mozilla Firefox, Microsoft Edge and main game developers support mjpg, the rest can be handled with add-ons.

- Minimum hardware is required due to the low computational density [8].

Video for Linux-2 (V4L2) server module has been used for image transfer from the camera via HTTP or HTTPS (HTTP-secure) protocols. V4L2 supported a web interface that allows to view the video stream and control the camera settings precisely and also supported basic authentication functionality for normal and administrative users.

Transmission Control Protocol/Internet Protocol (TCP/IP) Structure has been used as a set of rules governing communication over a network. These protocols define the movement of data between the source and the target or the Internet.

While TCP allows the data to be split into packets before transmission and reassembling them properly at the receiving point, the underlying layer IP controls the delivery of packets to the desired and correct address. TCP service is ensured by sockets with IP address and port number. TCP has been chosen as IP traffic, because of that;

- HTTP supports TCP.

- TCP is connection-oriented; after a connection is established, the data can be sent bidirectionally.

- TCP is suitable for applications requiring high reliability and the transmission time should be expected to be relatively less critical.

- There is a guarantee that the data transmitted on the TCP will remain intact and in the order in which they are sent.

- TCP performs Flow Control and Error Control/Correction. TCP requires a three-way handshake to establish a socket connection before any data is sent. Before a client tries to connect to a server, the server must first connect to the connection point to open connections: this is called passive open. After a passive open setup, a client can initiate an active open connection. When the host (server) sends a SYNchronize packet to the client computer, the client receives the SYN and sends a SYN-ACKnowledgement notification. The server also receives the SYN-ACK and finally declares the ACK. Thus, a TCP socket connection is established [9].

3. MOTION-JPEG STREAMING OVER HTTP

Motion-JPEG video streaming is supported by almost all browsers. This has been preferred because of widespread support, the easy obtaining from every camera and also the convenience it provides in operations. In the meantime, "video for Linux" module is used for mjpg streaming over HTTP under TCP/IP standards.

3.1. General Video/Image Transferring Logic

On the server, an array named "jpegImage" is defined for the byte type. Thanks to Golang, a second channel running parallel with the main code flow has been created with "go func ()" [10] and each picture taken from Picamera was assigned to the array in sequence and one at a time.

With the command "HTTP.listenAndServe ()", the HTTP server is started for incoming calls to the specified address and port (8080). "HTTP.Handle.Func ()" is activated with the incoming request from client and adds a header information to the jpegImage file to create a package that can be retrieved by the clients. The server then uses the TCP service to stream the jpeg format images over HTTP in sequence.

As long as the client wants to get the new jpeg and the server wants to provide, the TCP connection will not be closed and the image transfer will continue uninterruptedly. The algorithm of general video/image transferring logic is as shown in Figure 4.

The clients get;

- A jpeg image with HTTP://sunucu IP:8080/jpeg,

- Video/mjpeg with HTTP://sunucu IP:8080/mjpeg.

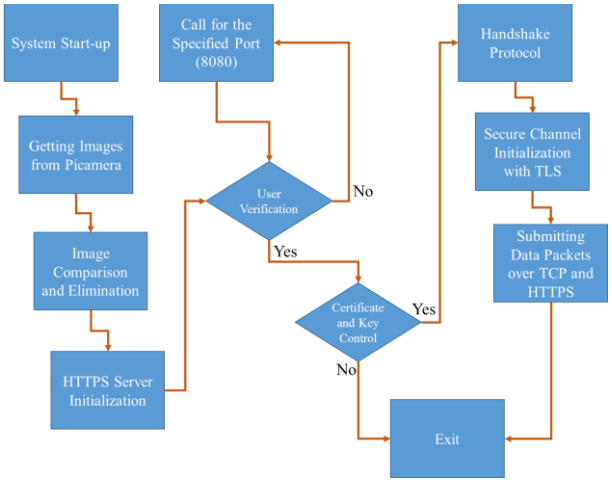

Figure 4. Algorithm

In the representative environment; more than one client (computer, tablet or smart phone) can receive images or video simultaneously from web browser or VLC player, and also image/video can be transferred to internet after making the necessary modem settings. The image can be viewed by clients in near-real-time and in an uninterrupted way with a delay of about 1 second.

3.2. Improving the Image Transfer with Blocking Duplicate Image Transmission

In the proposed method, it has been determined that there is a difference between the number of images transferred over the network and the number of images taken from the camera depending on the resolution of the image. While the camera does not exceed 60 frames per second (fps), the number of images transmitted over the network can reach to 1.200 depending on the resolution.

With an improvement made in the code, it was granted that only the new frames from the camera to be sent instead of sending the same picture for several times.

In the applications/experiments, 30 fps is taken as basis and the quantity data obtained from them are compared and presented in Table 1. As seen in table; while the number of transfers over the network in high-resolution images is reduced by half (1/2), at low resolution this ratio has been reduced to nearly one fortieth (1/40). The images at resolutions 2592x1944, 2368x1656 and 2144x1368 are not provided by the Raspberry Pi Camera Module, though they are claimed as features.

With this development made in the software,

- Sending of the same image more than once is prevented, so the number of pictures transferred and data traffic in parallel is also decreased in similar way,

- The use of CPU (Central Process Unit) and hardware necessity is reduced,

- The number of client connections that can receive the streaming is increased depending on the capacity of the network.

Table 1. Transferred image quantity and data comparison

Resolution	Before Development		After Development		Differences	
	(fps)	(MB/s)	(fps)	(MB/s)	(fps)	(MB/s)
1920 x 1080	60	21,67	30	10,80	30	10,87
1600 x 900	70	19,75	30	8,46	40	11,29
1280 x 720	75	15,45	30	6,18	45	9,27
960 x 540	80	10,70	30	4,01	50	6,69
640 x 360	90	6.76	30	2,25	60	4,51
320 x 180	150	3,66	30	0,73	120	2,93
160 x 90	450	3,59	30	0,23	420	3,36
80 x 45	1.200	3,51	30	0.08	1,17	3,43

4. CREATING AN ENCRYPTED CHANNEL FOR DATA TRANSFER

The processing of video files results in a large volume of data and increases the time required for coding by the software and hardware requirements.

Since the video system and the aircraft have to share limited battery and technical equipment, the encryption scheme should be designed as simple as possible as it also creates serious incompatibility between large video data and limited resources. This results in compromises from security. For this reason, instead of encrypting the data separately, in the proposed method for the UVAs, it is preferred to secure/encrypt the channel as a whole and to transfer the data/image/video via this secure channel to the ground control station. The benefits of this method are:

- To eliminate the need for extra equipment by using the restricted hardware (flight computer, video system, battery, etc.) located on the air vehicle and not to exceed the lifting capacity of the UAV or to reduce the flight time,

- Instead of encrypting the data to be received from more than one camera and the other flight data (like telemetry) separately, it is possible to transmit all the data over the channel that has already been encrypted and secured,

- To keep the hardware requirements at the minimum level in the ground control station as well as the air vehicle and to send commands to the air vehicle through the created secure channel,

- To reduce the time required to encrypt the data separately in the two-way (mutual) data transmission, thus eliminating possible delays in image transmission and thus vital errors (coordinate-based target detection, firing on target, etc.).

For the reasons explained above, it has been preferred to use Transport Layer Security (TLS) protocol to create a secure channel.

4.1. Certificate

Key and certificate files have been created for TLS protocol: 'cert.pem' and 'key.pem'. The certificate is signed by itself and is in X.509 standard. However, if desired, a certificate issued by a Certificate Authority (CA) may also be obtained and used.

Privacy Enhanced Mail (PEM): a method of securing e-mail using the public key cryptography developed in 1993. It is still widely used for storing key and X.509 certificates [11].

X.509: an important standard used in cryptography to manage digital certificates and public key cryptography. It is an important part of the TLS protocol used to secure web and email communications. X.509 as a standard determines the formats for public key certificates, certificate revocation lists, attribute certificates and certification path validation algorithm [12].

4.2. User Authentication and Authorization

It is guaranteed that each user is given separate access authority to view the image/video. Again, in order to ensure security, the creation of a new user account via the interface is not allowed. For user authentication and authorization, the database in the server holds the information in Table 2.

Table 2. User information

User Name	Password	Name, Surname
icuhadar	icuhadar1	İsmet ÇUHADAR
mdursun	mdursun1	Mahir DURSUN
user	password	-

After user names and passwords sent by the client to the server at the beginning of the session, the server verifies this user data and it is intended that the username and password are encrypted and sent over the secure channel, which is TLS, so that it cannot be obtained by third people. User authentication and authorization steps can be seen in Figure 5.

Image transfer is ongoing as long as the user has not exited and is not turned off by the server. Even if the client is the same, user name and password are requested again at every session opening/renewing.

It is aimed to keep the number of users under control by not allowing the creation of a new user account through the interface so that only the users determined in the database on the server can receive the image/video transmission.

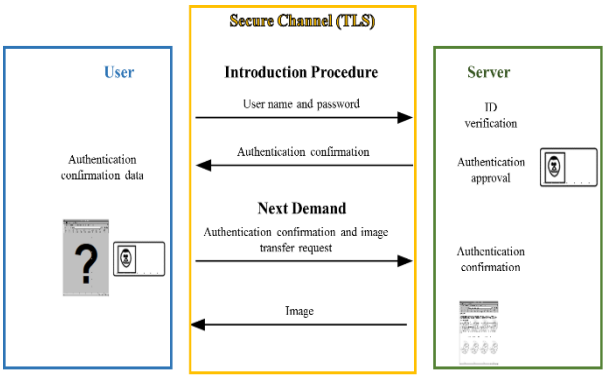

Figure 5. User login system [13]

In some Denial of Service attacks, attackers can make connections to the server, send any information until the time-out period, and increase the number of similar connections, which can unnecessarily waste server resources. Preventing such attacks and ensuring the efficient use of server resources is the result of initiating image transfer with only the correct username and password entry [14].

No matter how many clients and how many users are logged in, all entries and exits (session opening and closing) are recorded on the server with the time (date and time) and IP address.

After authenticating the user name and password, the server stores this information in a randomly generated cookie and uses this cookie to refer to it as session data. The information of authentication done is only kept in the server database for security concern. The page opened after the user's successful login is shown in Figure 6.

Figure 6. Page opened after user's successful login

5. CONCLUSION

In order to develop a secure image/video transfer method for UAV data link, a proper streaming method has been developed on the representative environment prepared with Raspberry Pi 3 and Raspberry Pi Camera Module. For this study, the Go programming language is preferred because of its advantages mentioned before.

Support by web browsers such as Safari, Google Chrome, Mozilla Firefox and Microsoft Edge, minimal hardware requirements, usage extensively by video capture devices

such as digital cameras, IP cameras and webcams made Motion-JPEG chosen as a powerful tool.

TCP protocol has also been used as IP traffic thanks to its proses as HTTP support, link-orientation, being suitable for applications requiring high reliability, guaranty that the data will go in unbroken and in the order in which it is sent and also Flow Control-Error Control/Correction.

Although various alternatives including 1080p can be used, image/video transmission is fixed with 30 fps based on 960x540 resolution. The number of pictures transferred is reduced by more than half by the above-described method, so the data traffic is also reduced. Thus, image/video transfer server has been developed which ensure clear and sharp close-to-real-time transferring with multi-client connectivity.

It has been preferred to use the TLS protocol because of that it encrypts more data and transfers it in a shorter time with low performance and hardware necessity and that it can provide bi-directional data flow securely.

With the secure/encrypted channel created with TLS, the Telemetry data (speed, altitude, engine speed and other flight information) can be send beside the images/videos from multiple cameras on the UAV.

The user login system has been created with an interface for the encrypted channel input and output to be secure. It is intended that the username and password are encrypted and sent over the secure channel (TLS) so that they cannot be obtained by eavesdropping. Furthermore, to be able to login to the system; user name and password comparison and verification is needed. Comparing the IPs authorized with the requesting IPs and the coordinates of the client requesting the connection with the known coordinates of the ground control station will be able to be compared.

In other studies encountered in the literature, video encryption is taken as the main purpose but the data link security is left as secondly. Because of this, the security of the data link to which the image/video is transferred and the procedure of user authentication/authorization are often ignored. Our method, on the other hand, is based on end-to-end security and it is also possible that the wireless data transmitted can be used not only for the aircraft but also for unmanned land and sea vehicles.

Whichever method is used, it is also important to use software such as firewalls and antivirus programs for general cyber security and if possible to have no internet access for UAV systems.

As a future work for developing proposed method it is possible to add this additional abilities; development of a new encryption method, storing hash values of user names and passwords in the database, password length and confusion check, two-stage authorization with one-time password (SMS, e-mail, etc.), renewal of user authority, name and password for certain period, connection time limits and lastly a panic button that allows all connections to be closed at the same time.

REFERENCES

[1] Robert, E., Guyot, V., Loica, A., Antoine. G., Laurent, B. (2015). Swarm UAV Attack: How to Protect Sensitive Data?. European Conference on Information Warfare and Security (ECIW). Laval, France.

[2] Şahingöz, Ö.K. (2013). Multi-level Dynamic Key Management for Scalable Wireless Sensor Networks with UAV. Springer Science and Business Media Dordrecht. Ubiquitous Information Technologies and Applications. DOI: 10.1007/978-94-007-5857-5_2.

[3] Rodday, N. (2015). Exploring Security Vulnerabilities of Unmanned Aerial Vehicles. Master's Thesis. Faculty for Electrical Engineering. University of Twente. Amsterdam. July 2015.

[4] Chen, X., Lifeng, W., Mengjiao, Z., Wemdong, W. (2016). A Resource-efficient Multimedia Encryption Scheme for Embedded Video Sensing System Based on Unmanned Aircraft. Journal of Network and Computer Applications. 59.117-125.

[5] Çuhadar, İ., Dursun, M., "Unmanned Air Vehicle System's Data Links", Journal of Automation and Control Engineering Vol. 4, No. 3, June 2016.

[6] Internet: Go programming language. https://en.wikipedia.org/wiki/Go_(programming_language), 22.12.2016.

[7] Internet: Senkovych, O. Golang webcam library for Linux. https://github.com/blackjack/webcam, 22.12.2016.

[8] Internet: Motion JPEG. https://en.wikipedia.org/wiki/Motion_ JPEG, 22.12.2016.

[9] Internet: TCP vs. UDP. http://www.diffen.com/difference/TCP_vs_UDP, 22.12.2016.

[10] Internet: The Go Programming Language, https://golang.org/, 22.12.2016.

[11] RFC 1421: Privacy Enhancement for Internet Electronic Mail.

[12] RFC 4158: Internet X.509 Public Key Infrastructure.

[13] Fu, K., Sit, E., Smith, K., Feamster, N. (2001). Dos and Don'ts of Client Authentication on the Web. MIT Laboratory for Computer Science.

[14] Sarıkaya, K. (2012). Password-based Clıent Authentıcatıon for SSL/TLS Protocol Usıng Elgamal and Chebyshev Polynomıals. Master's Thesis. Hacettepe University, Department of Computer Engineering.