Indonesian Journal of Electrical Engineering and Computer Science
Vol. 10, No. 2, May 2018, pp. 756~762
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v10.i2.pp756-762

Securing Data Communication for Data Driven Applications Using End to End Encryption

Subhi Almohtasib, Alaa H Al-Hamami
Department of Computer Science, Princess Sumaya University for Technology, Jordan

Article Info

Article history:

Received Nov 6, 2017
Revised Jan 26, 2018
Accepted Feb 11, 2018

Keywords:

Adversary
Cryptography
Decryption
Encryption
Smartphones

ABSTRACT

Many users of smartphones have secret data they want to save it on their devices. The probability of a device damage or stolen prevents them from saving data. Therefore, data driven applications used to save user's data on a remote server. Protection of the data during its transmission considered as one of the success aspect for these applications. In this paper, an enhanced method for data encryption proposed which guarantees data secrecy during its transmission over network. User's data encrypted before transmission using Base64 class. Data encryption and decryption implemented to halt reverse encryption process. In this way, data is transmitting in a secure and efficient manner accomplishing the main goal of Cryptography.

Corresponding Author:

Subhi Almohtasib,
Department of Computer Science,
Princess Sumaya University for Technology,
Amman 11941 Jordan.
Email: Subhi.Almohtasib@gmail.com

1. INTRODUCTION

Nowadays, the smartphones usage around the world has been exceeded one billion users at the end of 2012 and it is expected that the next billion devices could be reached within the next five years [1]. Lightweight and less cost of mobile devices paved the way for people to keep secret data on their mobile devices. Thus, Hiding and preventing data from accessed by unauthorized people is important to users.

Although saving user's data on the same device with a password to restrict access to these data is a good way of protection, but users still have fears about password being compromised; which means users will lose the privacy of their data. Moreover, if the device has stolen or damaged, user will lose his/her data forever. Because of that, using data-driven application is very useful for users to save their significant data.

Data driven applications used to keep data on an online storage; where users can access their data anytime and anywhere they want. To ensure secrecy of data, it has to be unreadable for unauthorized people at any time during data transmission process and this done using cryptography techniques. Cryptography is the study of constructing ciphers with the help of an effective encryption method to ensure the confidentiality and integrity of data. Thus, preventing the data from manipulated while transmitting and hiding it from unauthorized people are the major aspects of success of the encryption method [2].

In general, Cryptography consist of two processes Encryption and Decryption. Encryption is the process of encoding messages in such a way that only the authorized parties can read it. That means the encrypted data will be in a different form from the original data. While Decryption is the process of decoding the encrypted text into the primordial text. One of Cryptography methods is Symmetric key encryption method. Using this method, a special key used to encrypt data and it requires the same special secret key to decrypt data. The input to the symmetric encryption method will be the user's data that he wants to secure

usually called plaintext. Moreover, the output of the method is the cipher text, which will be unreadable text to anyone.

All information that passes over computer network sent in packets. For instance, when an E-mail is sent from one Computer to another, first it broken up into smaller segments. Each segment called a packet and it has the destination address, the source address, and other information such as the number of packets and reassembly order of packets. Once they arrive at the destination Computer, the packets reconstructed again.

TCP/IP protocol defines how data sent and received over network. Such a protocol guarantees the data integrity using two error-checking methods, the Cyclic Redundancy Check (CRC) and Checksum, to verify that data have not altered during transmission. These methods cannot hide the content of packets through its trip over network that means the content of packets would be vulnerable to detected by an adversary [3].

Day by day adversaries are trying to find weakness or insecurity in the cryptographic methods in order to know and alter the content of the detected packets. Packet sniffing is the process of capturing all packets passed over the network and looking for any information that may be useful [4]. Most of the time, packet sniffing tools are used to troubleshoot network problems like finding out why traffic is too slow in one part of the network, but this doesn't stop hackers to use these tools to disclosure the packets content. One of generic attacks to networks is the Passive Attack; using this attack the adversary can read the content of the messages without altering the message itself. A popular type of passive attack is Man-In-The-Middle (MITM) attack; which done by an adversary during packets communication between a client/server model. It gives the adversary the ability to detect and disclosure packets content through communication. Recently, the huge grown in usage of data driven applications had increased concerns for users of being victims of MITM attack. In this paper, a new End-to-End encryption approach provided to increase the secrecy of data during transmission from the mobile device to server.

When encryption method applied to a certain data, it must have done with a manner that data can be decrypted easily. Encryption methods that use a secret key in generating ciphertext from the plaintext and uses the same secret key to generate the plaintext again refers to Symmetric encryption. The major subject of symmetric encryption is how to share the secret key between sender and receiver. Thus, using a predefined method for both sides that uses the same-shared key makes encryption and decryption more secure. The Exclusive OR operation used widely in the symmetric encryption due to its ability to regenerate plaintext.

Researchers had provided a variety of methods for symmetric encryption. In [5], the authors proposed a new mechanism for encryption and decryption but it has to be done on the server side in order to reduce power computations on the smartphone and save its battery from being drained.

The encryption method provided by the authors starts with converting the plaintext to its equivalent ASCII character array and each character position found. After running a random function between the arrays positions, the corresponding value of each random number inserted in new array, this constitutes the secret key. Now, Ciphertext generation is done after the values of the ASCII array are XOR-ed with the values of the random array. Finally, Ciphertext and the secret key sent to the smartphone and ciphertext shown to user. Different steps performed on Decryption process; it requires ciphertext and secret key to send from the smartphone to the server. Next, the ciphertext and secret key are XOR-ed and the result is the original plaintext (original ASCII text), now method starts to get values of each 3 digits together, finally the value from the ASCII text is sent again to the smartphone which is displayed to the user.

Despite the proposed method by the authors guarantees data integrity because it uses TCP/IP protocol's error check methods, but it does not guarantee data confidentiality in two cases. The first one, while data transmitted from the smartphone to the server, after data decrypted, and sent back to the smartphone. The second one, if the adversary XOR-ED the captured ciphertext and secret key then he/she can get the ASCII array for plaintext. Although, this methodology reduces the computational power for the smartphone and saves its battery since no computations done on the smartphone; but this is not secure in terms of security. Since the transmitted packets captured and filtered using packet sniffing software and the message privacy violated.

In [6], the backpropagation neural networks have been used to create the private key. The formation of the key is done by taking the weights of from the input text to hidden layers in the used network. Later on, the final key is taken from the outputs to that network. The used step function is the sigmoid function and with 10 hidden layers. Despite this approach can generate a key for any length of input, but it may be trapped by the instability of the hidden networks. Which cannot be accurate all of the time. Thus, it might encrypt a plaintext but it won't be able to decrypt it.

In [7], Algebraic chess notation used to encrypt and decrypt data or chess mapping. This approach based on the chessboard feature that it has 64 squares and each of the alphabet and numerals are assigned with a specific square of the chessboard. But since the chess mapping is a well-known algorithm and if an

adversary detects the transferred packets, and using any simple hacking application which will test the chess mapping technique and other techniques., then the confidently is violated.

The authors presented another technique in [8], they had replaced the 3D message character matrix with a 2D matrix to enhance the security. This called the message masking or character masking and it depends on the dynamic value of each character taken from the input. But the problem that it encrypts only 99 characters.

2. RESEARCH METHOD

End-to-End encryption means that data will encrypted before transmitted over the network then to decrypt it at the receiver device. This way makes it harder to disclosure the original data. In addition, data encryption before sending it to the server will increase data confidentiality [9]. Before proceeding with the proposed approach, the following section shows how to use packet detection tools to implement MITM attack and message content disclose.

2.1. Exisiting Methodology

To detect the transmitted packets over network the adversary can use Wireshark software, Wireshark is an open source network packet analyzer software that captures the network packets and then display the packet data in detail. Wireshark is widely used to analyze the network traffic, to find the loopholes in the network architecture. In order to detect some of the attacks on the network and to provide solutions for them. Sometimes this software does not detect all the transmitted packets because not all packets headers are available in the Wireshark database headers. Therefore, adversaries usually use RawCap software with Wireshark to capture all packets that are passing over the network. RawCap it is also works a packet detection software but with different packet headers that Wireshark use [10], [11].

Now, both software constituting a good detection platform for packets passing over a certain network, then the attacker knows the content of each captured packet. Figure 1 shows the data that user wants to encrypt. For example, suppose that the user wants to secure this data" PRIVATE DATA", the attacker can detect the packets that contain the user's data as shown in Figure 2.

Figure 1. Entered Plaintext.

In the other hand, the existing methodology sends the ciphertext and secret key from the server to the smartphone after encryption method applied to the ASCII value. In general, adversaries use hacking software, which will help them to predict the used encryption scheme; most of these hacking software checks the value of the Decimal XOR operation [12] [14].

Figure 2. Packet Detection using Wireshark Software.

Table I shows the Decimal XOR-ED value of secret key and the ciphertext for different rounds of encryption for the same data. Supposing that the attacker had detected many packets for the same plaintext. This might help attacker to decrypt data, this refers to pattern attack that gives the attacker the ability to determine the type of the encryption and decryption scheme in the symmetric key encryption.

Table 1. Key Retrieval Using XOR

Secret key	Ciphertext	XOR-ED value
01001070110009	10010676144141	1000971160970330
1900090	4277575251	64035049
71090119101019	99553562129836	1000971160970330
1170000	786740025	64035049
11010001791111	11050730959356	1000971160970330
11009911	4136518174	64035049
91117990901917	98574001874210	1000971160970330
19177010	897635803	64035049

2.2. Proposed Methedology

The proposed methodology keeps the existing encryption and decryption methods at the server side; but it encrypts the data before sending it to the server. If Man-in-the-Middle attack occurs, this way prevents the adversary to understand the message content and the message will be irreversible. A symmetric encryption done on the smartphone using Base64 class encoding Algorithm. Base64 is available for JAVA programming language. Base64 encoding a group of similar binary-to-text encoding schemes that represent binary data in an ASCII string format by translating it into a radix-64 representation [13]. Figure 3 Shows packet detection for same plaintext but after it had been encrypted using Base64 encoding class.

Data encryption and decryption will have done on two phases, the first phase on the smartphone using the Base64 class and sent the encrypted text to the server, the second phase is done on the encrypted

text from phase one but on the server which uses the original encryption method. Data encryption and decryption will have done on two phases, the first phase on the smartphone using the Base64 class and sent the encrypted text to the server, the second phase is done on the encrypted text from phase one but on the server which uses the original encryption method.

The displayed data for the user are the double encrypted plaintext as it shown in Figure 4. Now, for decryption process, the doubled encrypted plaintext sent back to server and decryption method for the server applied on it. This will generate the encrypted text at phase one, after that this text sent to the smartphone and another decryption method is applied, which will result the user's plaintext.

Figure 3. Packet Detection After Encryption Using Base64 Class.

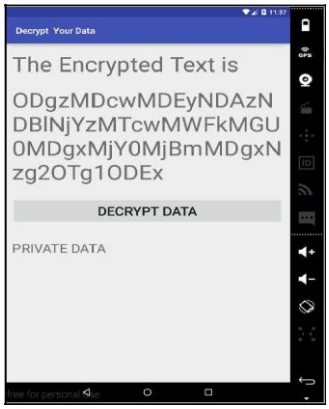

Figure 4. Displaying Ciphertext and Plaintext After Decryption Method Done

Since the key is generated randomly then it will be harder to the adversary to know the pattern of key generation because inputs are varying. Plus, if the same input presented to the algorithm, the random function will generate different key at each time of execution. Thus, confusion is guaranteed.

3. RESULTS AND ANALYSIS

The implementation of the proposed approach has done on a virtual device created by Genymotion software, which is equivalent to the real smartphone. An application of the proposed approach must have downloaded on the virtual device, and it requires an internet connection to send and receive data with server. The user has to enter the desired data to secure as it shown in Figure 1.

The powerful of the proposed methodology lies in two aspects. Firstly, because data encryption and decryption done on two phases one at the smartphone itself and the other on the server; this would be a challenge for the adversary to know the content of the detected packets. Secondly, after the encryption done at phase two, the returned text to smartphone is the ciphertext only; i.e. without its secret key. This will prevent the pattern attack using the Decimal XOR. Table II shows the same encrypted text as received from phase one encryption and the different secret key, ciphertext after phase two encryption is applied.

Table 2. Generated Ciphertext by the proposed approach

Smartphone Encrypted Text	Secret Key	Ciphertext
UFJJVkFU RSBEQVRB	8060000664774887 6410012806600220 2266000700907877	ODgzMDcwMDEyNDAzND BlNjYzMTcwMWFkMGU0 MDgxMjY0MjBmMDgxNzg 2OTg1ODEx
UFJJVkFURS BEQVRB	028471800901 858498784560 0036686086 60568800060010	MGFkNDAxODc0OTc1OGR lNTlmN2Y0NWU1MDgxNm ViNjZlNjA5NWU5ODg2MG UyMDc2
UFJJVkFURS BEQVRB	687408200330 7695650848780 63868832797013 050600677	NjAyNDc4Mjc0MzQ0N2Vm NDYyMGY0OGZkMGUxO GViODU0N2ZlMDkyMGQ2 NjgyNjEx
UFJJVkFURS BEQVRB	00800088006082 06060075584703 30006580008807 700170	MDhkMDcwOGY0MDE0OG E2NzAxMDc3NWRkNGYy M2IzMDYwNWU5MDg5OD gxNzgyMTE2

4. CONCLUSION

Data protection is an important field when online applications are used. Adding the End-to-End encryption scheme to these applications will make it harder for adversaries to disclosure the detected data. In addition, it makes it more secure against reversible cryptography techniques. The proposed methodology protects the users from being victims of pattern attack. Nevertheless, users can save their private data online without concerning about secrecy of data. Users of end-to-end encryption applications would have no fears about saving their personal data in a secure manner on their portable devices.

5. FUTURE WORK

Many applications are demanding the multimedia transmission between users; the proposed approach applied to image and voice data, the user given the ability to upload the required data to encrypt. Furthermore, applying the End-to-End encryption scheme to other devices like standalone computers and servers or any machine that communicates with other machine, will have a great impact to the newest trends in communication for example, the Internet of Things. Since it needs fast and reliable encryption techniques.

REFERENCES

[1] D. R. Selvarani and T. N. Ravi, "A Review on the role of Encryption in Mobile," *International Journal of Application or Innovation in Engineering & Management*, vol. 3, no. 12, pp. 76-83, 2014.

[2] A. J. Amalraj and J. J. R. Jose, "A SURVEY PAPER ON CRYPTOGRAPHY TECHNIQUES," *International Journal of Computer Science and Mobile Computing*, vol. 5, no. 8, pp. 55-59, 2016.

[3] P. B. Nath and M. Uddin, "TCP-IP Model in Data Communication and Networking," *American Journal of Engineering Research*, vol. 4, no. 10, pp. 102-107, 2015.

[4] Y. Ketkar, W. Khan, D. Makwana, V. Nemade and A. Hutke, "A Protocol Based Packet Sniffer," *International Journal of Computer Science and Mobile Computing*, vol. 4, no. 3, pp. 406-410, 2015.

[5] S. Bhagoliwal and J. Karjee, "Securing Mobile Data using Cryptography," *The International Journal of Advanced Networking and Applications*, vol. 7, no. 6, pp. 2925-2930, 2016.

[6] R. S. D. Y. J. Neeru Rathee, "A Novel Approach for Cryptography Using Artificial Neuarl Networks," *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 4, no. 4, pp. 187-193, August 2016.

[7] V. M. C. A. G. R. P. A. Manimaran, "Encryption and Decryption Using Algebraic Chess Notations," *International Journal of Pharmacy & Technology*, vol. 8, no. 4, pp. 22099-22105, 2016.

[8] S. N. P. O. K. P. Dinesh P. Baviskar, "ANDROID BASED MESSAGE ENCRYPTION/DECRYPTION USING MATRIX," *International Journal of Research in Engineering and Technology*, vol. 4, no. 1, pp. 315-320, 2015.

[9] M. Pertierra, D. Garcia and A. E. Lugo, "Messengr: End-to-End Encrypted Messaging Application with Server-Side Search Capabilities," Massachusetts Institute of Technology, Massachusetts, 2016.

[10] J. Gehring, "Packet Analysis Using Wireshark," Florida Gulf Coast University, Fort Myers, Florida, 2011.

[11] C., "RawCap and Wireshark: How to capture and analyze local traffic from host machine to itself," cmd - developing software, 11 January 2014. [Online]. Available: http://carminedimascio.com/2014/03/rawcap-and-wireshark-how-to-capture-and-analyze-local-traffic-from-host-machine-to-itself/. [Accessed 22 April 2017]

[12] A. Kaminsky, M. Kurdziel and S. Radziszowski, "*An overview of cryptanalysis research for the advanced encryption standard*," in MILITARY COMMUNICATIONS CONFERENCE, MILCOM, 2010.

[13] J. Platform, "Class Base64.Encoder," 01 January 2013. [Online]. Available: https://docs.oracle.com/javase/8/docs/api/java/util/Base64.Encoder.html. [Accessed 22 April 2017].

[14] M. abdel-qader, A. Al-Jaber and A. Al-Hamami, "Using Short Message Service (SMS) to Support Business Continuity," *World of Computer Science and Information Technology Journal (WSCIT)*, vol. 1, no. 2, pp. 34-38, 2011

BIOGRAPHIES OF AUTHORS

Subhi Almohtasib received his BS in Computer Science with high honors Palestine Polytechnic University, Palestine in 2016. He is currently a student of Computer Science at Princess Sumaya University for Technology, Jordan. Mr. Almohtasib is interested in Database and Knowledge-Base Systems, Algorithms, and Automated Software Engineering.

Ala'a Al-Hamami received his BS in Physics from University of Baghdad, Iraq in 1970 and an MS in Computer Science from University of Loughborough Technology, England in 1979. In 1983, he received his Ph. D from the University of East AngliaGeorge Mason, England. He is a Professor of Computer Science at Princess Sumaya University, Amman, Jordan. Prof. Al-hamami is interested in Computer Security, Computer Networks, and Internet of Things.