VOL. 10, NO. 18, OCTOBER 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences
©2006-2015 Asian Research Publishing Network (ARPN). All rights reserved.

www.arpnjournals.com

SECURITY ENHANCEMENT OF ADVANCED ENCRYPTION STANDARD (AES) USING TIME-BASED DYNAMIC KEY GENERATION

Zuhar Musliyana[1], Teuku Yuliar Arif[2] and Rizal Munadi[2]
[1]Department of Informatics Engineering, Faculty of Computer Science, Ubudiyah Indonesia University, Jalan Alue Naga, Desa Tibang, Banda Aceh, Indonesia
[2]Wireless and Networking Research Group (Winner), Department of Electrical Engineering, Faculty of Engineering, Syiah Kuala University, Jalan Tgk. Syech Abdurrauf No.7 Darussalam, Banda Aceh, Indonesia
E-Mail: zuhar@uui.ac.id

ABSTRACT

Login is the first step which is conducted every time we access to a system and only granted to those who are entitled. Login is very important and it is a part of the security of a system. Registered user or guest and administrator are two kinds of users that have the privilege and access. Nowadays, wrongdoers are always on the dark side and frequently try to gain access to the system. To provide a better security, it is important to enhance the access mechanism and to evaluate the authentication process. Message-Digest 5 (MD5) is one of the algorithms that commonly used in the login system. Although it has been so popular, but the algorithm is still vulnerable to dictionary attacks and rainbow tables. In addition to the hash function algorithm, Advanced Encryption Standard (AES) algorithm alternatively can be the best choice in the login authentication process. This study is proposed to develop a dynamic key generation on the AES algorithm using the function of time. The experimental results obtained that AES key can be generated at random based on the value of the time when a user logs in with a particular active period. In this implementation, the security authentication becomes stronger because of changes in key generating chipertext changes for each encryption process. Based on the time as a valuable benchmark, the result shown that AES encryption-decryption process is relatively fast with a time average about 0.0023 s.

Keywords: AES, dynamic key, time-based, chipertext, login.

INTRODUCTION

Today, the login authentication system became a major issue in a variety of web-based applications to secure the privilege access. Login system is an important part of becoming a barrier between the client and administrator access in managing a content application. Among the algorithm alternatives, Message-Digest 5 (MD5) is often used. The using of The MD5 algorithm is still shown vulnerable to dictionary attacks and rainbow tables [1]. In addition, collusion also became a serious issue that must be considered if we still want to use this algorithm [2]. There are several algorithms hash functions such as MD5, Secure Hash Algorithm 1 (SHA1), SHA256 and so on that can be used in the login authentication. Besides that, there are also some other alternatives that become the best option. For example the using of a symmetric algorithm such as Data Encryption Standard (DES), AES, International Data Encryption Algorithm (IDEA), Rivest Cipher 4 (RC4), or asymmetric algorithm such as Rivest Shamir Adleman (RSA), Digital Signature Algorithm (DSA), Diffie Hellman (DH), Elliptic Curve Cryptography (ECC).

In the past, solutions were implemented to improve the security of the login system in reducing attacks of cryptanalysis,can be done by boosting the size of the encryption key. This solution is not the best, because a larger size of key does not guarantee the password cannot be decrypted. Each of cryptography key is safe only for a specified time. Related research [3] confirms that 1024 bit RSA encryption used in most banking and e-commerce system only safe for a few years. In addition to that, the larger size of key often requires a higher power source for the computing process. The large size of the encryption key is also not suitable for mobile devices that lack of resources and limited battery size. On the symmetric algorithms such as AES key is determinant in performing the encryption decryption. To generate a strong encryption, key generation takes a dynamic concept, but this key must be recognized when the process description.

This study uses the value of time as the key generation for AES algorithm that written in the PHP programming language. The key will be generated by taking the value of the time when a user logs in to the system. On the decryption process synchronization takes a time value with a certain tolerance limits to find the same key pair of the time value generated in the encryption process. Application of this time limit of tolerance can produce stronger encryption because the key can only be used at certain times to perform encryption decryption.

LITERATURE

Cryptography: Cryptography is a science that studies how data is converted into a specific design that difficult to be understood [2]. Cryptography aims to maintain the confidentiality of information or data that cannot be identified by unauthorized parties (unauthorized person). Data can be encoded as plaintext or cleartext. Furthermore, the data that has been encrypted called ciphertext. In cryptography required parameters used for the data conversion process that is a set of keys.

©2006-2015 Asian Research Publishing Network (ARPN). All rights reserved.

Encryption and decryption of data are controlled by a key or some keys [4].

Advanced encryption standard algorithm: AES is a symmetric cryptographic algorithm that is secure enough to protect data or confidential information. In 2001, AES is established as the latest standard cryptographic algorithm by the National Institute of Standards and Technology (NIST) as a replacement for the Data Encryption Standard (DES) algorithm [5]. The AES algorithm can encrypt and decrypt the data with variable key length are 128 bits, 192 bits, and 256 bits [5].

Encryption: The AES encryption algorithm has a process that consists of 4 types of transformation bytes, i.e. SubBytes, ShiftRows, Mixcolumns, and AddRoundKey [5]. In the first step of the encryption process, the input on the state will sustain transformation of AddRoundKey bytes. The next stage, the state will experience a transformation of SubBytes, ShiftRows, MixColumns, and AddRoundKey repeated as much as Nr. This process is called a round function. The final round will have a difference with the previous round of the last round, but this state no suffered a MixColumns transformation [5]. The AES encryption algorithm can be seen in the Figure-1.

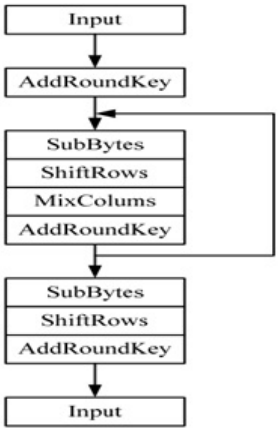

Figure-1. Flow diagram encryption of AES [7].

Decryption: This module takes between 128-bit blocks of data, performs AES (AKA Rijndael) decryption with a user-entered 128-bit key. The results of this process are stored in the SRAM [6]. In AES decryption process, cipher transformation can be reversed and implemented in the opposite direction to produce the inverse cipher to be easily understood. Transformation of bytes that used in the inverse cipher for AES decryption process are I, InvSubBytes, InvMixColumns, and AddRoundKey.

METHODOLOGY

Flowchart: The design of dynamic key generation may be indicated in the following flowchart diagram:

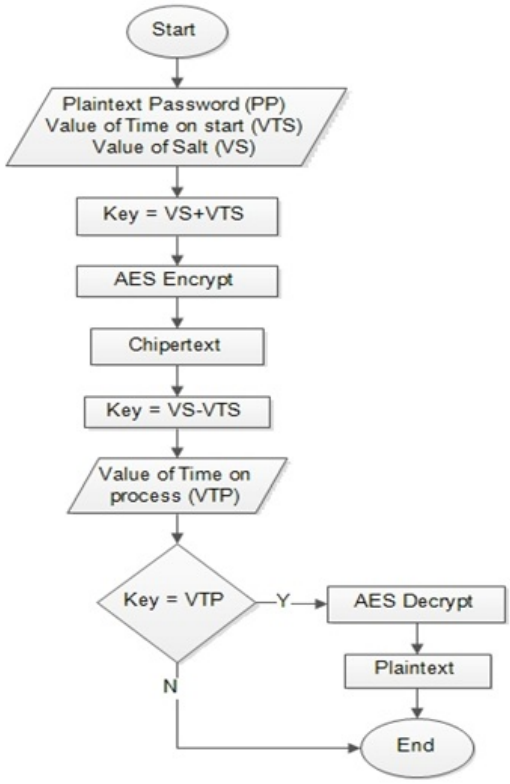

Figure-2. Flowchart of dynamic key generation.

Figure-2 above explains the stages of the process on the encryption algorithm AES with dynamic key generation. The value of salt (VS), plaintext passwords (PP), and value of time on start (VTS) are data for the initial input that will be used in the encryption process. Before the encryption process, VS and VTS have been mixed before. Furthermore, before the decryption process, VS and VTS are separated key to get the key that contains the value of time without a value of salt. After that, key matches with the value of time to process (VTP) to get the key for decryption process.

Pseudocode: The pseudocode environment requires the fancy box package by Timothy Van Zandt. This package is described in Section 10.1.3 of [8]. A dynamic key generation on AES algorithm is implemented by taking the value of time when the user login. The first step is to set the time zone so that the value of time when the key is generated in sync with the current time value of the key used in the decryption process. The pseudo code of dynamic key generation on AES can be seen in the Figure-3.

©2006-2015 Asian Research Publishing Network (ARPN). All rights reserved.


```
Initialization
password, key, time, salt : string
time ← get_time
input ← (password)
key ← salt + time
Encryption
chipertext ← AES encrypt (password, key)
output(chipertext)
Decryption
key ← salt - time
for as much tolerance given time
    if key = get_time
        key ← salt + time
        plaintext ← AES decrypt (chipertext, key)
    end if
end for
output(plaintext)
```

Figure-3. Pseudocode of dynamic key generation.

Based on the pseudocode above, the value of time taken, subsequently mixed with value of salt to reinforce the key that be generated, and stored as the value of a variable in the form of a hash value. Furthermore, this variable will be used as the key for the AES encryption-decryption algorithm. On the decryption process, AES is a symmetric algorithm that must use the same key for encryption decryption. In the encryption process, keys generated dynamically utilizing a combination of salt and the value of time. Therefore, at this step it is necessary to synchronize the value of time to get the same key when the encryption process is done. In the pseudocode above, an iteration is used to find the value of time when encryption is done as much tolerance time given. This tolerance is to provide a check on the possibility of space delay between the time when a user logs in with the user validation login. It also illustrates that AES keys have active certain period after the user logs in.

One time password: One Time Password is an authentication method that uses the- always -changing-password after each login, or change at regular intervals [9]. OTP can be divided into two main categories based on mathematical algorithms and time-based synchronization. OTP method is based on mathematical algorithms using complex mathematical algorithms such as a cryptographic hash function in order to generate a new password based on the previous password and shared secret key [9]. While the second method is based on the type of time synchronization changes constantly in each unit of a certain time interval [10]. This process requires time synchronization between the client and the authentication server, for generating a new password based on the current time and not on the previous password or a secret key.

RESULT AND DISCUSSION

The picture below is display login form interface through a web browser.

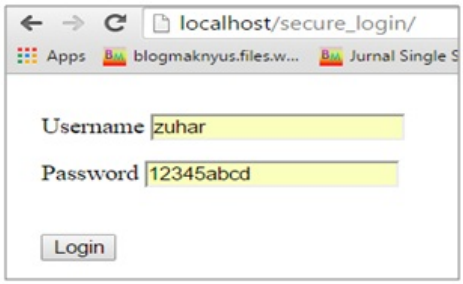

Figure-4. Form login.

In Figure-4 the above, experiment conducted by entering the input login with username: *zuhar* and password: 1234abcd four times. The examination results are shown in Figure-5.

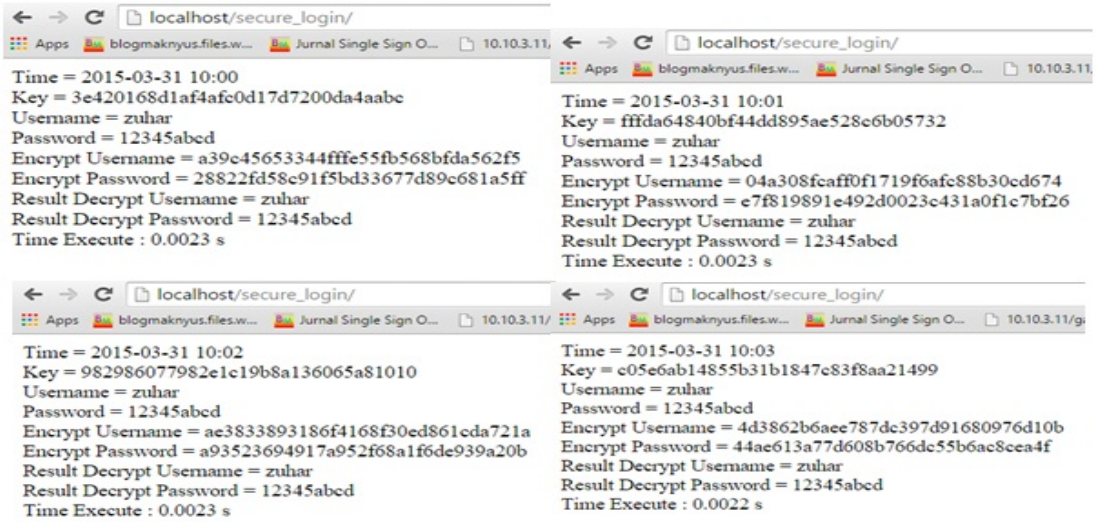

Figure-5. Login test with random.

©2006-2015 Asian Research Publishing Network (ARPN). All rights reserved.

In Figure-5 shown the AES key is randomly generated based on the value of time when the user logs in. This change also makes the chipper text key change for each encryption process. Based on the experimental results AES encryption-decryption on average takes about 0.0023 s.

CONCLUSIONS

Dynamic key generation in AES can be generated by utilizing a function of time by taking value of time when the user logs in, then the decryption process of synchronizing time with a certain tolerance to find the key of the value of time which is generated in the encryption process. The development of this dynamic key can generate stronger encryption key since it is generated due to the dynamic nature and it is only can be used at certain intervals. The experimental results also shown that the using of time in the process of AES encryption-decryption is relatively fast; with an average of 0.0023s.

REFERENCES

[1] M.C. Ah Kioon, Z. Wang and S.D. Das. 2013. “Security Analysis of MD5 algorithm in Password Storage,” Proceedings of the 2nd International Symposium on Computer, Communication, Control and Automation (ISCCCA-13).

[2] Alahmad M.A., Alshaikhli I. and Jumaah B. 2013. "Protection of the Digital Holy Quran Hash Digest by Using Cryptography Algorithms," Advanced Computer Science Applications and Technologies (ACSAT), 2013 International Conference on ,Vol. No., pp. 244, 249, 23-24 December doi: 10.1109/ACSAT.2013.55.

[3] Huy Hoang Ngo, Xianping Wu, Phu Dung Le and Campbell Wilson. 2010. Balasubramaniam Srinivasan, “Dynamic Key Cryptography and Applications”, Faculty of Information Technology, Monash University 900 Dandenong Road, Caulfield East,Victoria, 3145, Australia.

[4] Ravi Ranjankr 2012. Symmetric Encryption by Symmetric Algorithm Classes[Online].Available:http://www.codeproject.com/Articles/442523/Cryptography-Symmetric-Encryption-by-Symmetric-A, August.

[5] Sahoo O.B., Kole D.K. and Rahaman H. 2012. "An Optimized S-Box for Advanced Encryption Standard (AES) Design," Advances in Computing and Communications (ICACC), International Conference on ,vol., no., pp.154,157,9-11Aug.2012 doi: 10.1109/ICACC.2012.35.

[6] Shrivathsa Bhargav, Larry Chen, Abhinandan Majumdar and Shiva Ramudit. 2008. "FPGA-based 128-bit AES decryption," CSEE 4840 - Embedded System Design Spring, Columbia University.

[7] Fei Shao, Zinan Chang and Yi Zhang. 2010. "AES Encryption Algorithm Based on the High Performance Computing of GPU," Communication Software and Networks, 2010. ICCSN '10. Second International Conference on , vol., no., pp. 588,590, 26-28 February. doi: 10.1109/ICCSN.124.

[8] M. Goossens, F. Mittelbach and A. Samarin. 1994. The LATEXCompanion, Addison-Wesley.

[9] N. M. Haller. 1994. “The S/KEY TM one-time password system”, Proceedings of the Internet Society Symposium on Network and Distributed Systems, pp. 151- 157.

[10] S. Gupta, S. Sahni, P. Sabbu, S. Varma and S. V Gangashetty. 2009. "Passblot: A Highly Scalable Graphical One Time Password System," International Journal of Network Security & Its Applications (IJNSA).