*International Journal of Engineering & Technology*, 7 (4.41) (2018) 126-128

International Journal of Engineering & Technology

*Website: www.sciencepubco.com/index.php/IJET*

***Research paper***

# A Tokenization System to Secure Critical Data

**Chetana Pujari [1]*, Balachandra Muniyal [1], Aishwarya Kulkarni [2]**

*[1] Department of Information & Communication Technology,*
*Manipal Institute of Technology,*
*Manipal Academy of Higher Education, Manipal, Karnataka, INDIA-576104*
*[2]SAP BASIS,*
*Atos India Pvt Ltd*
**Corresponding Author Email: chetana.pujari@manipal.edu*

**Abstract**

One of the recent developments in India is a major shift from cash to cashless transaction which is going to bring in drastic change in the world of e-commerce. As transaction is a crucial task, the need for security is the prominent factor to consider. The proposed work aims to revise AES and develop a secure system based on tokenization.This method ensures no relation between actual credit card data and the token generated for it, so all the applications will be dealing with token but not the actual data. In tokenization there is no need to store token and corresponding data in a database instead a standard random value table is used to generate token, which reduces the overhead of storing tokens and ensures reduction in security audit.

***Keywords***: *POS, token, cryptography, security, critical data, cipher, AES*

## 1. Introduction

Over the past few years, India has seen continual growth in the number of online transactions. People prefer paying their bills online. There are around 700 million credit/debit cards in circulation in India right now. Demonetization and the go cashless initiatives have led to significant growth in number of digital payments [4]. To promote cashless transactions, the Indian Government has initiated a program called Digital India, with an idea to convert India into a powerful digital society. Since the government is also encouraging online transactions, providing enough cyber security is essential to sustain the incredible growth the industry is experiencing.

### 1.1. Cyber security and its necessity

Currently trend in Indian is to make the country corruption and fraud free, which requires people to go cashless, which means we need to use credit card but credit card usage is not very popular in India as people are not convinced about its security, hence there is a need to ensure security. The proposed system aims at developing such secure system as e-commerce merchants deals with token instead of credit card data, even if the hackers attack their system, hacker will acquire only valueless tokens. Providing security for any transaction is a prominent factor not only that the proposed system ensures low cost and high performance, which is much needed in today's cashless world.

### 1.2. Problem Statement

Low cost, high security and high performance are the prominent requirements for any e-commerce applications; the proposed system aims to ensure them. As the card data will be stored in centralized secure repository and all e-payment application will be dealing with token, the security audit will be reduced to great extent hence low cost and high performance. Once the user is convinced about the security of his/her card usage, the transition from cash to cashless will happen smoothly, which in turn reduces black money, corruption and fraud to great extent. The proposed system can be further extended to provide security for any confidential data like aadhar card details etc.

## 2. Background

### 2.1. What is tokenization?

Tokenization is the process of replacing sensitive data elements with non-sensitive equivalents, referred to as tokens. Tokens have no utilizable meaning as it is independent of original data. There is one to one mapping between token and original data. Special method is used to do mapping from original data to token which make it impossible to reverse the token back to the original data. The tokenization system must ensure that all security services like confidentiality, integrity, authenticity and availability are guaranteed.

### 2.2. Where is tokenization used?

Tokenization can be used for any crucial application like online banking, ecommerce or retail application, online voting system, sensitive medical data and subtle data related to business applications[8].

### 2.3. Why is tokenization used?

When Tokenization is used, instead of storing customer's credit card data at e-commerce sites, the site stores token so even though if someone hacks ecommerce site, hackers may get hold of only tokens which are independent of actual data. Mapping tokens to credit card data will be a tedious task for hacker. The data security

risk is considerably reduced by tokenization and is a comparatively easy way to regulatory compliance. Tokenization ensures minimum exposure to crucial data. The necessities of the PCI DSS are simplified by employing tokenization, as PCI DSS systems does not hold sensitive data.

### 2.4. AES Algorithm

Advanced Encryption Standard(AES) is one of the popular encryption algorithms, which is also known as Rijndael cipher. AES is based on the concept of substitution and permutation. In AES a block size is fixed to 128 bits and key size of 128, 192, or 256 bits can be used. The key size corresponds to the number of rounds that converts the given plain text, into its corresponding cipher text. The count of cycles of reiterations are 10, 12, 14 for 128, 192, 256 bit keys respectively. Decryption uses same key to get back original data.

## 3. Literature Survey

Jayshree Katti et al. [1] have proposed a system to overcome the drawbacks of the already existing security systems. AES was implemented in the client side as an effective approach to provide security to transmitting and storing data. On implementing AES, a significant increase was observed in the time taken to encrypt and decrypt files, and to upload an encrypted file.

Fang Rao et al.[2] has proposed an improved version of AES, as there was a little possibility of global optimization in AES, they chose to local optimization by modifying the MixColumn transformation of AES. On implementing this modified version, they observed that not only the code size reduced, but the overall energy consumption of ZigBee networks was reduced. A similar result is expected with the proposed version of the revised AES algorithm.

Jun Shu et al.[3] have proposed version of AES, a resource sharing fast encryption and decryption algorithm. The s-boxes and inverse s-boxes are replaced with a look-up table. A resource sharing is done in encryption and decryption and the area of hardware was also effectively reduced. In order to share & protect resource the series-mix transformation of AES was enriched. This approach further optimized the algorithm and also reused key computing components. This version of the algorithm achieved more frequency and throughput, reached a better balance and had better performance.

Danushka Jayasinghe et al.[6] have proposed EMV tokenization. This approach enables offline transaction by implementing the protocol that ensures end to end encryption of the data for security reasons.

Shreya Paul et al.[7] have proposed tokenization model for cloud environment, for the crucial data like health care data, transaction data and any confidential data related to business, an additional level of security is ensured by generating token[10].

## 4. Design

### 4.1 System design

The figure, Fig.1 depicts the overall design for a revised version of AES, revised is used to encrypt the cardholder's data. The tokenization system takes the encrypted text as input and returns a randomly generated number referred to as token. The merchant/ POS terminal stores the token instead of the actual data or the encrypted data[11].

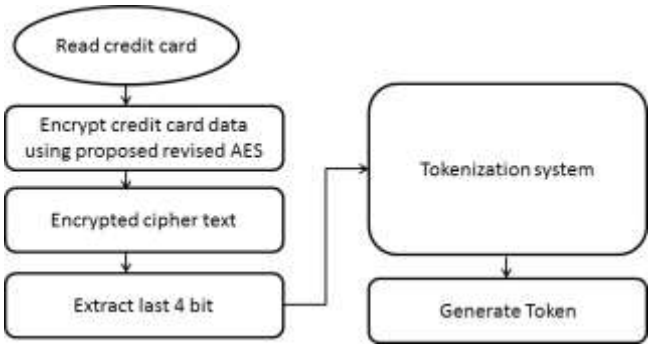


**Fig.1:** System design

### 4.2. Design for revising AES algorithm

When a transaction is done online, the cardholder's data cannot be freely exchanged between servers, it has to be encrypted. AES algorithm is one of the best encryption algorithms. The proposed system aims at improving the response time by revising byte substitution layer in each round of AES.

The 16 x16 S Box used in the substitution layer of AES encryption algorithm is replaced by simple lookup table in the proposed system.

| | | | |
|---|---|---|---|
| $Y_{0,0}$ | $Y_{0,1}$ | $Y_{0,2}$ | $Y_{0,3}$ |
| $Y_{1,0}$ | $Y_{1,1}$ | $Y_{1,2}$ | $Y_{1,3}$ |
| $Y_{2,0}$ | $Y_{2,1}$ | $Y_{2,2}$ | $Y_{2,3}$ |
| $Y_{3,0}$ | $Y_{3,1}$ | $Y_{3,2}$ | $Y_{3,3}$ |

**Fig.2:** Input to lookup table

Figure Fig.2 represents input state to lookup table, where each cell represents 1 byte whose value is within GF(2^8)(Galois Field). The lookup table consists of two columns, the first column represents state value and the second column represents corresponding substitution byte for each state.

The figure Fig.3 represents proposed lookup table where $B_i \in GF(2^8)$, where $0 \leq i \leq n$, the lookup table is sorted with respect to its first column. When the state is given as input to the lookup table, each byte in a cell is searched in first column of lookup table using binary search algorithm and is replaced with corresponding substitution byte. The result of the lookup table is stored as new state and then given as input to the shift row layers of AES, figure Fig.4 represents the output state.

Inverse substitution is performed by using inverse lookup table which is obtained by swapping first and second column of lookup table as shown in Fig.5.

| | |
|---|---|
| $B_0$ | $B^1_0$ |
| $B_1$ | $B^1_1$ |
| $B_2$ | $B^1_2$ |
| . | . |
| . | . |
| . | . |
| $B_n$ | $B^1_n$ |

**Fig.3:** Lookup table

| | | | |
|---|---|---|---|
| $Y^1_{0,0}$ | $Y^1_{0,1}$ | $Y^1_{0,2}$ | $Y^1_{0,3}$ |
| $Y^1_{1,0}$ | $Y^1_{1,1}$ | $Y^1_{1,2}$ | $Y^1_{1,3}$ |
| $Y^1_{2,0}$ | $Y^1_{2,1}$ | $Y^1_{2,2}$ | $Y^1_{2,3}$ |
| $Y^1_{3,0}$ | $Y^1_{3,1}$ | $Y^1_{3,2}$ | $Y^1_{3,3}$ |

**Fig.4:** Output of substitution layer

| | |
|---|---|
| $B^1_0$ | $B_0$ |
| $B^1_1$ | $B_1$ |
| $B^1_2$ | $B_2$ |
| . . . | . . . |
| $B^1_n$ | $B_n$ |

**Fig.5:** Inverse lookup table

### 4.3 Design of the Tokenization System

The tokenization system has a token database called the token vault which assigns tokens to the encrypted data. The original data and the tokens have no mathematical relation, so that there is no risk even in the case of a breach.
There is no other way to trace the tokens back to the original, except for having access to the token vault. Below is a flowchart shown in figure Fig.6 of the working of the tokenization system with references to the overall design as shown in figure Fig. 1.
The tokenization system works in way that the same token is generated when the same data is sent to be tokenized. Each card number, for instance, should generate a unique token.

Consider the following example. Let's say the card C1 is used in a transaction at time t1, and generates a unique token k1. The tokenization system has to make sure that if the same card C1 is used in a transaction at time t2, the same token k1 should be generated. Suppose another card C2 is used and generated token k2, for the tokenization system to be secure and valid, it is necessary that the tokens k1 and k2 are not the same.

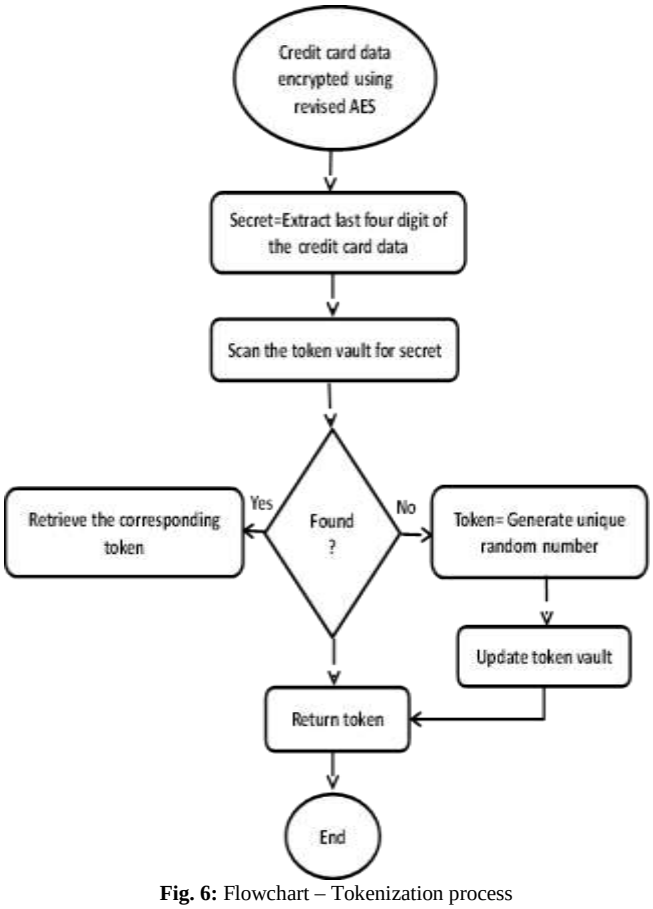


**Fig. 6:** Flowchart – Tokenization process

## 5. Conclusion

AES is one of the most effective encryption algorithms. Using AES to encrypt cardholder data ensures that the risk is minimal. The tokenization system is implemented to ensure and additional level of security. The tokens generated are sent to the merchant and are used in the transaction instead of the actual card numbers. The merchant can also store the tokens for any further transactions such as recurring payments and refunds. By storing only the tokens and not the actual card data, the risk of data leak in case of an attack in reduced significantly.

## References

[1] Niteen Surv, Balu Wanve, Rahul Kamble, Sachin Patil, Jayshree Katti (2015), "Framework for client side AES encryption technique in cloud computing", 2015 IEEE International Advance Computing Conference (IACC).
[2] Fang Rao, Jian Jun Tan, "Energy consumption research of AES encryption algorithm in ZIGBEE", International Conference on Cyberspace Technology 2014, IET, Bejing China.
[3] Jun Shu, Yiwen Wang, Wenchang Li, Zhiyong Gan (2010), "Realization of a resource sharing fast encryption and decryption AES algorithm", 2010 International Symposium on Intelligent Signal Processing and Communication Systems December 6-8,2010.
[4] "By 2025, digital transactions in India could be worth $1 trillion a year",Available:https://economictimes.indiatimes.com/industry/banking/finance/banking/by-2025-digital-transactions-in-india-could-be-worth-1-trn-a-year/articleshow/63284898.cms, March 13, 2018.
[5] "Encryption as a Service (EaaS) as a Solution for Cryptography in Cloud", The 4th International Conference on Electrical Engineering and Informatics, 2013.
[6] Danushka Jayasinghe, Konstantinos Markantonakis, Iakovos Gurulian, Raja Naeem Akram and Keith Mayes, "Extending EMV Tokenised Payments To Offline-Environments", 2016 IEEE TrustCom-BigDataSE-ISPA.
[7] Shreya Paul, Atma Prakash Singh and Shafeeq Ahmad, "Tokenization Based Service Model for Cloud Computing Environment", 2016 International Conference on Inventive Computation Technologies.
[8] Ram Kumar Garg, NK Garg, "Developing Secured Biometric Payments Model Using Tokenization", 2015 International Conference on Soft Computing Techniques and Implementations, India.
[9] Noor Ashitah Abu Othman, Fakariah Hani Mohd Ali and Mashyum Binti Mohd Noh, "Secured Web Application Using Combination of Query Tokenization and Adaptive Method in Preventing SQL Injection Attacks", 2014 IEEE 2014 International Conference on Computer, Communication, and Control Technology, Malaysia.
[10] Sandra D´ıaz-Santiago, Lil Maria Rodriguez-Henriquez and Debrup Chakraborty, "A Cryptographic Study of Tokenization Systems", International Journal of Information Security 2016.
[11] PCI Security Standards Council: Payment Card Industry Data Security Standard Version 1.2 (2008). https://www. pcisecuritystandards.org/security_standards/pci_dss.shtml