May – June 2020
ISSN: 0193-4120 Page No. 5584-5590

Providing Advanced Security for Card Validation Using Enhanced Luhn Algorithm in Cloud Computing

V. Sireesha[1], M. Usha Rani[2]
[1]Research Scholar, Department of Computer Science SPMVV, Tirupati.
sireesha.vankana@gmail.com

[2]Professor, Dept. of Computer Science, SPMVV, Tirupati.
musha_rohan@yahoo.com

Article Info
Volume 83
Page Number: 5584 - 5590
Publication Issue:
May - June 2020

Article History
Article Received: *19 November 2019*
Revised: *27 January 2020*
Accepted: *24 February 2020*
Publication: *17 May 2020*

Abstract
The most important security issue in cloud computing which is disturbing the organizations is data protection, because they will not transfer their data to remote machines if there is no guarantee about data protection from the cloud service providers. One of the sensitive issue that is related to data protection is card data (credit, master, visa etc.,) protection. The existing techniques/algorithms for card data security have undergone some failures regarding the confidentiality and integrity matters. Sometimes, the sensitive data or information generated in unencrypted format to other systems resulted in unauthorized access and disclosure. Based on these reports, in this paper we presented an IBM Cloud Hardware Security Module (HSM) encryption and tokenization-based system for credit and debit card information security. The proposed system consists of the HSM and card processing modules as well as Enhanced Luhn Module (ELM).The ELM is a luhn based card validation system, which can compute and process card transactions in restricted and more securely in cloud. The implementation of the system was carried out on Intel core i7 9th generation processor and 8 GB of RAM on Red Hat Enterprise Linux 7.7 Operating System. TOMCAT server and HTML with CSS JavaScript served as the frontend with JDK 1.7 for business processing and MySQL as a database. Analysis of the results of implementation with Master, Amex and Visa cards showed that the system delivered very high usability and more secured experience for users.

Keywords: *Luhn algorithm, checksum, card validation, cloud computing, tokenization,*

I. Introduction

"Day-by-day the users of Internet in cloud are growing rapidly due to the consequent growth in Information Technology (IT) industry in recent years. As the Internetoffers great opportunity for improved communication and experience to the users globally, it has also provided new drives for credit/debit-card-based businesses and marketing by many financial institutions towards reaching out to new markets and creating opportunities for economic growth. The card payment process model is shown in the figure below. In this process, the customer submits the relevant information such as card number, payment fee and expiry date and so on to the merchant for verification [3]. The merchant bank verifies the information and submits to the Payment Gateway (PG). The PG passes the received information to our system ELM, which validates the card and establishes a linkage with the customer's bank for a decision on whether to accept or drop the request. The amount is not transferred immediately to the merchant's bank, but instead, a token of settlement, which is a merchant's electronic payment information for a certified transaction, is delivered [1]. The proliferation of Internet-based

May – June 2020
ISSN: 0193-4120 Page No. 5584-5590

businesses has before time been largely attributed to the rising confidence levels and trust among participants and the extent to which information confidentiality can be maintained. However, in view of the emerging range of fraud, theft, disruption and denial of service attacks on online transactions, stakeholders have expressed great concern, doubt and loss of confidence on credit/debit card information transmission, privacy, integrity and security [10]."

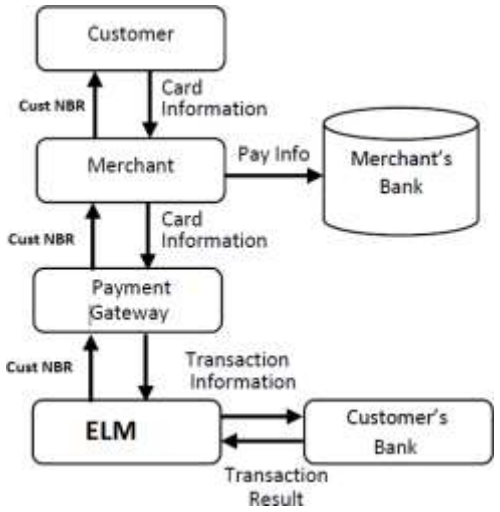

Fig 1. Card payment process model

"With the existing and traditional approaches for card systems, there are some security issues like unauthorized access, information theft and other breaches that leads to money loss, which in turn leads to insecure feeling in customers [1]. Generally a credit/debit card fraud is committed when an individual inadvertently and callously use the information on the card for deception, misrepresentation and other selfish and personal reasons. It will happen through lost/stolen card, account takeover, Cardholder-Not-Present (CNP), magnetic stripe erasure, card counterfeiting and skimming. The most commonly used techniques in Internetfor credit/debit card fraud include site cloning, false merchant sites and unlawful card generation. Some other frauds arise from merchant side, as if the merchant and employee collide each other to do fraud using customers' accounts details and personal information. The risks associated with credit/debit card fraud often pared together on the merchant and the cardholder, who may be unaware of the attack. While cardholders are faced with the daunting challenge of getting a fraud-related charge reversed, merchants are confronted with sales lost arising from payback fees as well as threat of account closure or suspension [15]. Therefore, there is a need to provide more secure techniques to develop the trust in customers in using the card. Here we discussed about the enhanced card validation technique that ultimately provides more security than previous methods."

II. Security Methods for Card Processing

"For providing the security to card data, a number of methods have been suggested which are discussed here. During the transmission process in card data, to prevent eavesdropping the Secure Socket layer (SSL)-based encryption has been used .This method depends on asymmetric key encryption for customer-merchant communication and promotes digital certificate-based authentication of the identity of the merchant [11]. Another protocol, Secure Electronic Transaction (SET) has been proposed for ensuring secured credit card payment over the Internet. It works by establishing protocols for cardholder and merchant registration, purchase request, payment authorization and payment capture [7]. With dual signature concept, SET prevents illegal use of credit card number by enforcing the exclusive sharing of information on transaction order and payment information with the merchant and bank respectively. However, this method is more complex which often results in some incidences of card insecurities over the Internet. Some other algorithms like encryption and cryptography had been introduced to provide more security for card information. The Encryption algorithm transfers the card information into a form that makes it impossible to read without the appropriate knowledge (a key). It provides privacy by hiding the information from intruders and impostors. Further, by using decryption the encrypted data can be converted back into readable format that is required. Cryptography is classified into secret key and public key [8]. Secret-key cryptography is also known as symmetric cryptography and works with same key for encryption and decryption as shown in fig. 2 and

May – June 2020
ISSN: 0193-4120 Page No. 5584-5590

the most popular ones are Data Encryption Standard (DES), Advanced Encryption Standard (AES), Triple DES, Blow Fish, Two Fish, Three Fish among others [4]."

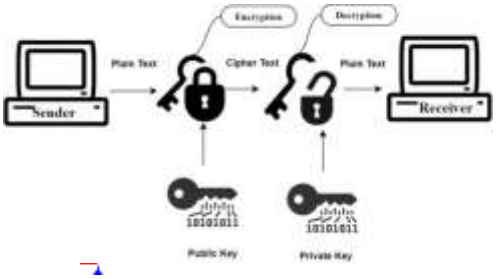

Fig 2. Public key cryptography

"Public-key cryptography is also known as asymmetric cryptography. Here for encryption two different keys are used, one public and one private key as shown in Figure 3. The public key is published and associated with users in a trusted manner, while the private key is made secret based on non-sharing of information between the sender and receiver thereby promoting privacy and authentication [5]. Common public key algorithms include Elliptic Curve Cryptography (ECC) and RSA algorithms [6]."

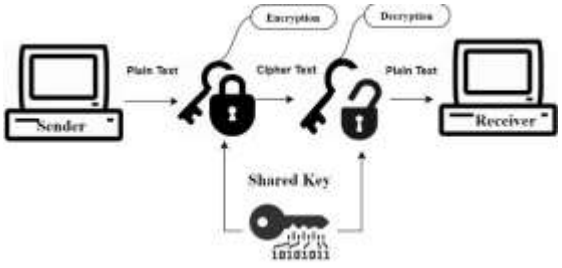

Fig 3. Private Key cryptography

III. Proposed System

"The proposed system consists of the HSM, card processing modules and ELM as a sub-module, which addresses some of the limitations of the previous works and provides more security for card data, whichare conceptualized in figure below in a cloud environment."

Card Swipe Machine

"A card swipe machine or a point of sale terminal (POS terminal) is an electronic device used to process card payments at retail locations [14]. It reads the information of a customer's credit or debit card and sends the information to Card processing System, which is present in the cloud. Point of sale terminals are a combination of software and hardware that allows retail locations to accept card payments without updating their cash registers to read cards directly."

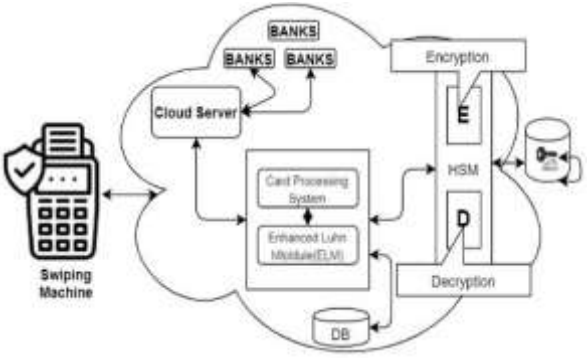

Fig 4. Architectural design of the proposed system

Card Processing System (CPS)

"In this system, the process and validation of the card information takes place in different steps as discussed below:

1. The CPS (card processing system) using the Cloud server API, reaches to different banks and authorise the card information.
2. After completing the Initial Authorisation phase, CPS process the card information for payment process.
3. Under payment process, CPS process card details using different techniques
    a. CPS Initially take the Original card Number and encrypt the card number using the IBM HSM.

TEST Engineering & Management

May – June 2020
ISSN: 0193-4120 Page No. 5584-5590

Account number	7	9	9	2	7	3	9	8	7	1	**x**
Double every other	7	**18**	9	**4**	7	**6**	9	**16**	7	**2**	**x**
Sum digits	7	**9**	9	4	7	6	9	**7**	7	2	**x**

b. The encrypted card data that is created looks same as Original Card Number for example,

Original Card Number: 1234 5678 9101 1121

Encrypted Card Number: 1234 **** **** **21

c. HSM always generates the same unique token for each Card number that initializes it. This way a token can be resubmit to a HSM in order to retrieve the original PAN in an encrypted format and the same keys are used for decryption.
d. After completing the encryption process, CPS send the encrypted card data to Enhanced luhn module to generate the checksum using the Enhanced luhn calculationor modification for more security.
e. The Encrypted card data will need to be saved in a database outside of the HSM. Since the card number is encrypted using strong encryption methods this will be compliant with PCI DSS, which can be used for future use.

4. After completing the card processing, the response is send back to the POS terminal,which transfers the funds from the customer's account to the seller's account (or at least accounts for the transfer with the credit card network), records the transaction and prints a receipt."

- **Enhanced Luhn Module (ELM):**

"The Luhn algorithm or Luhn formula, also known as the "modulus 10" or "mod 10" algorithm, is a simple checksum formula used to validate a variety of identification numbers, such as credit card numbers, IMEI numbers, National Provider Identifier numbers in the United States etc [9].

Assume an example of an account number "7992739871" that will have a check digit added, making it of the form 7992739871x:

The sum of all the digits in the third row, the sum of the sum digits, is 67.

The check digit (x) is obtained by computing the sum of the sum digits then computing 9 times that value modulo 10 (in equation form, ((67 × 9) mod 10)). In algorithm form:

5. Compute the sum of the sum digits (67).
6. Multiply by 9 (603).
7. 603 mod 10 is then 3, which is the check digit. Thus, **x=3**.

(Alternative method) The check digit (x) is obtained by computing the sum of the other digits (third row) then subtracting the units digit from 10 (67 => Units digit 7; 10 − 7 = check digit 3). In algorithm form:

1. Compute the sum of the sum digits (67).
2. Take the units digit (7).
3. Subtract the units digit from 10.
4. The result (3) is the check digit. In case the sum of digits ends in 0, then 0 is the check digit.

This makes the full account number read 79927398713."

Pseudocode implementation

```
function checkLuhn(string purportedCC) {  int
sum:=integer(purportedCC[length(purportedC)-1])
int nDigits := length(purportedCC)
   int parity := nDigits modulus 2
for i from 0 to nDigits - 2 {
      int digit := integer(purportedCC[i])
if i modulus 2 = parity
         digit := digit × 2
if digit >  9
digit := digit - 9
      sum := sum + digit
   } return (sum modulus 10) = 0
```

May – June 2020
ISSN: 0193-4120 Page No. 5584-5590

```
}
```

"The algorithm is in the public domain and is in wide use today. It is specified in ISO/IEC 7812-1. It is not intended to be a cryptographically secure hash function; it was designed to protect against accidental errors, not malicious attacks [12]. Most credit cards and many government identification numbers use the algorithm as a simple method of distinguishing valid numbers from mistyped or otherwise incorrect numbers. To overcome these limitations and to provide more security, we have enhanced the Luhn algorithm to calculate the checksum by padding number one to the calculated checksum."

The pseudocode implementation for ELM to calculate the checksum for Encrypted Card data is as follows:

```
function checkLuhn(string purportedCC) {
   int sum :=
integer(purportedCC[length(purportedCC)-1])
   int nDigits := length(purportedCC)
   int parity := nDigits modulus 2
for i from 0 to nDigits - 2 {
      int digit := integer(purportedCC[i])
if i modulus 2 = parity
         digit := digit × 2
if digit > 9
          digit := digit - 9
            digit:= digit+1
      sum := sum + digit
   } return (sum modulus 10) = 1
}
```

- **Luhn validation Phase**

"In this validation phase, we can find whether our proposed luhn check has been applied or not. If the luhn check completes and if it returns 0 as the check digit, then normal luhn or mod 10 has been applied which is formally a valid number. If it returns 1 as a check digit, then our enhanced luhn check has been applied, which is a more secure method, and the card is a valid one else the card is invalid. The code to validate the card using ELM is given below."

```
function doValidate() {
    let pan =
document.getElementById("PAN").value;
    pan = pan.split(" ").join("");
    let sum = -1;
    let format = new RegExp("^[1-9][0-
9]{10,18}$", "g");
    if (format.test(pan)) {
            sum = luhnSum(pan);
            let tmp = pan.substr(0, 4);
            for (let i = 4; i < pan.length; i += 4)
                tmp += " " + pan.substr(i, 4);
            pan = tmp;
    }
    let text;
    switch (sum) {
case 0: text = "is a formally valid PAN."; break;
case 1: text = "is a formally valid ** Encrypted
PAN."; break;
default: text = "is not a valid PAN!"; break;
    }
    document.getElementById("PAN").value =
pan;
    document.getElementById("result").innerH
TML = text;
}
```

Hardware Security Module (HSM)

"The hardware security module (HSM) is a dedicated crypto processor that is specifically designed by IBM for the protection of the crypto key lifecycle. Hardware security modules act as trust anchors that protect the cryptographic infrastructure of some of the most security-conscious organizations in the world by securely managing, processing, and storing cryptographic keys inside a hardened, tamper-resistant device [9]."

"In this module, it receives the card number from the CPS and performs both encryption and decryption process which is discussed above by generating a token using tokenization concept.Here we used Triple DES technique for encryption. The keys are stored in the separate database, which is provided outside the HSM. The HSM always generates the same unique token for each Card number that initializes it. This way a token can be resubmit to a HSM in order to retrieve the original PAN in an encrypted format and the same keys are used for decryption."

May – June 2020
ISSN: 0193-4120 Page No. 5584-5590

- **Tokenization**

"Tokenization is the process of substituting a sensitive data element with a non-sensitive equivalent, referred to as a token, which has no extrinsic or exploitable meaning or value. The token is a reference (i.e. identifier) that maps back to the sensitive data through a tokenization system. The mapping from original data to token uses methods, which render tokens infeasible to reverse in the absence of the tokenization system, for example using tokens created from random numbers [2]."

Here the sensitive data is a PAN and the token generated is a non-sensitive equivalent.

"There are two variants for tokenization to consider and a decision should be made regarding which to use. The first alternative is to generate a token with a maximum length of 22 characters. This will be created by retaining the BIN number and then padding the remainder so that the total length is the 22 character limit including a new control digit so that the Luhn algorithms checksum equals 0. This option has the advantage that it is rather simplistic to differentiate in the system, which numbers have been tokenized and which have not, since no cards in the market have a length of 22 [13]."

"The other option is to retain the original card length so that the output matches the input length (16 characters for MasterCard, 13/16/19 for Visa). The BIN number will be retained. The remaining characters will be generated including a new control digit so that the Luhn algorithms checksum equals 0. This solution has the advantage of allowing us to send in the generated token into the current mainframe with little to no changes. Both methods for tokenization can be achieved with the IBM HSM.

The concept of tokenization requires that the HSM always generate the same unique token for each PAN that initializes it. This way a token can be resubmitted to a HSM in order to retrieve the original PAN in an encrypted format.

The Token-PAN combination will need to be saved in a database outside of the HSM. Since the tokenized PAN is encrypted using strong encryption methods this will be compliant with PCI DSS."

IV. Conclusion

"This paper presented the Enhanced Luhn Algorithm for card validation to provide more security fromdifferent frauds.Our system requires cloud-computing technology to function and its main advantages include its ability to ensure non-repudiation of transaction as well as secrecy of card transaction data or information. This system is tested especially for petroleum vendors by swiping cards, which resulted in more effectiveness, speed efficiency and applicability. It was also showed that the system would deliver very high usability, adaptability and favourable experience for users. Comparative analysis with related and relevant systems showed that our system gives 2 % extra security for cards.It is revealed that the proposed system showed better performance (99%) in securing credit/debit card information."

References

[1] Li Y., Zhang W.: Securing credit card transactions with one-time payment scheme, *4*, 413–426 (2005). https://doi.org/ 10.1016/j.elerap.2005.06.002, Accessed 12/05/2016.

[2] Sadeghi A., Schneider T., Winandy M., Horst G.: Token-Based Cloud Computing, Springer-Verlag Berlin Heidelberg, 417–429 (2010).

[3] Khan S. S., Scholar M. E.: Cloud Security Using Multilevel Encryption Algorithms, 5(1) (2016) https://doi.org/10.17148/ IJARCCE.2016.5116, Accessed 16/05/2017.

[4] Zhou M., Jiang Z.: Design and Implementation of Cloud Storage Security System, Applied Mechanics and Materials, 220, 2325-2329 (2012).

[5] G. Saini, and N. Sharma, Triple Security of Data in Cloud Computing, International Journal of Computer Science and Information Technologies (IJCSIT), 5(4), 5825-5827 (2014).

[6] Kakkar A., Singh M. L., Bansal P. K.: Comparison of Various Encryption Algorithms and Techniques for Secured Data Communication in Multinode Network, *2*(1), 87–92 (2012).

May – June 2020
ISSN: 0193-4120 Page No. 5584-5590

[7] Ismail R, and Zainab A. N.: Information systems security in special and public libraries: an assessment of status, *16(2)*, 45–62 (2011).

[8] Kartit Z., Marraki M. E. L.: Applying Encryption Algorithm to Enhance Data Security in Cloud Storage, Laboratory of Research in Informatics and Telecommunication (LRIT), University of Mohammed V, Faculty of Sciences, Rabat, Morocco (2015).

[9] Lin Q. T., Wang C. D., Pan J., Ling L., Lai J. H.: Local Data Security and Privacy Protection in Cloud Service Applications. *Ninth International Conference on Frontier of Computer Science and Technology*, 254–258 (2015). https://doi.org/ 10.1109/FCST.2015.39, Accessed 17/07/2017.

[10] Kunze M, Lizhe W, Jie T, Gregor VL. Cloud computing: A perspective study. Rochester Institute ofTechnology RIT Scholar Works; 2008.

[11] Tirthani N, Ganesan R. Data security in cloud architecture based on Diffie Hellman and elliptical Curve cryptography. IACR Cryptology ePrint Archive. 2014;49(5).

[12] Behera T. K., Panigrahi S.: Credit Card Fraud Detection: A Hybrid Approach Using Fuzzy Clustering andamp;amp;amp; Neural Network. *Second International Conference on Advances in Computing and Communication Engineering*, 494–499 (2015). https://doi.org/10.1109/, Accessed 18/02/2017.

[13] Rajak S, Ashok V. Secure data storage in the cloud using digital signature mechanism. International Journal of Advanced Research in Computer Engineering and Technology. 2012;1(4).

[14] Jeff H. Smart grids: Digital certificates and encryption play key role in security. WYSE Technology; 2012.

[15] Sivasakthi T, Prabakaran N. Applying digital signature with encryption algorithm of user authentication for data security in cloud computing. International Journal of Innovative Research in Computer and Communication Engineering. 2014;2(2).