

# Secured E-Banking services using CBM

JELPHINA LOBO, ZAHIR AALAM

**Abstract** — Using the concept of CBM (Confidential Building Metric) and OTP (One time password) this paper aims to present security for online banking transaction. To preserve the confidentiality of user details, a modified AES algorithm is used for Encryption. This algorithm will substitute byte, shift rows remains same as in the original AES while 128-bit permutation operation will replace mix column operation after adding round key operation. Certain parameters will be required to compute the CBM and can be implemented at the client side easily irrespective of any platform. In case of unsure transaction a time limited password can be generated to authenticate the user with the conceptualization of OTP. Different type of Threats and Attack scenarios that can take place any of the E-banking services or transactions have also been discussed in this paper.

**Index Terms** —CBM (Confidential Building Metric), OTP (One Time Password), Modified AES (Advanced Encryption Standard), Cryptography.

## I. INTRODUCTION

A form of banking where electronic signals are exchanged between financial institutions for the purpose of transferring money is known as E-banking. Customers are able to interact with their bank accounts as well as initiate any financial transaction from anywhere and at anytime.

The confidentiality of user details and bank information details is one of the biggest concern of the internet users. Bank users using online services of banking are always worried with the critical risk of someone attacking their system. Therefore, the authentication of the identity of both the sender and the receiver needs to be checked by a trusted third party who holds the identity certificates. To deal with the potential vulnerabilities regarding security and protection, many authorities have developed various solutions and systems. Still the security and privacy features concerning to the E-Banking services needs to improve to continue a successful, but secure growth. Due to the wide use of Electronic banking, many concerns have been risen from different perspectives: government, businesses, banks, individuals and technology. Banking institutes are emerging with a wide range of financial services to their customers. Banks are definitely concerned about the security of their system with a wide opening of financial transactions been processed on a daily basis. They may have to face a lot of serious damage and lost of capital, for the lack of security [1].

Now, lets discuss about the frequently found threads and the attack scenarios with regards to E-banking authentication. The approach let down by the authors of [14] has been presented in this section.

## A. THREATS

Serious threats related to authentication and authorization of users in the financial sector are identified, consisting a categorization of threats based on the component/asset attacked.

- 1) Threats against end-users:
  - a) Physical observation.
  - b) Social engineering.
  - c) Phishing.
- 2) Threats against end-users' devices:
  - a) Physical theft of token or device.
  - b) Tampering (or replication) of token or device.
  - c) Malicious software.
- 3) Threats against communication networks:
  - a) Pharming.
  - b) Eavesdropping, Interception and Hijacking.
- 4) Threats against remote banking services:
  - a) Code injections against web banking server:
  - b) Brute force attacks against online banking server.
  - c) Bank data breach.

## B. ATTACK SCENARIOS

This section aims to highlight, the attack methodologies targeting the E-banking services.

- 1) The High Roller Attack Scenario. In first stage of this attack [5], attackers used online tactics in order to gather information of the users, and send specifically crafted and highly targeted emails to bank customers with high balance accounts. Then the attacker users spear phishing to achieve the installation of their malicious code in the victim's node. As soon as the E-banking session is started, the malware is triggered. Now, this malware initiates a silent fraudulent transaction to an unknown account, while the user is still waiting for the authentication. It thus manages to hide such transaction from the legitimate user.
- 2) The Eurograbber Attack Scenario. The anatomy of the Eurograbber attack was the following [6]: firstly, the user's PC was infected with a variation of the Zeus Trojan malware, through a malicious email attachment or through a drive-by download. The malware remains in the stealth mode until the bank site is been accessed by the user. The malware convinces the user to provide

their mobile number, using social engineering techniques. Then, a link is been send through SMS to the user's cell, to download a malicious link, referring an application of security. With such an application installation, the user's PC as well as the mobile device is infected by the attacker. Thus, when the user login in to the bank account, a fraudulent transaction is initiated by the malware to a mule account, while the mobile receives an SMS code which is then send back to the attacker by this malware [2].

## II. OBJECTIVE AND PROBLEM DEFINITION.

Attackers have many ways and schemes that they can use to dissolve and form any system.

Following are the objectives of this research work:

- Create a system using Confidence building metrics to ensure secured financial transactions.
- To ensure secure end-to-end transaction to communicate over untrusted channels.
- To make the individuals understand the vulnerabilities of the online transaction systems and effectives methodologies for the same.
- Provide an enough adequate security to banking, security and e-commerce at our financial institutions.

Online banking is a concept that has been introduced to facilitate the process of paying bills instead of performing this by mail and deposit checks at bank branch. Each person can enter the bank website, choose online baking and insert his personal identification number (PIN) and password. This feature gives the clients the option of bypassing the time-consuming, paper-based aspects of traditional banking in order to manage finances more quickly and efficiently.

As mentioned, security is an important factor since it might be a turning point in the whole online banking principle through its threat to the bank's reputation. This means that the security attribute must reach a high level of reliability in order to thwart any attempts from the attackers to breach the system and steal its sensitive information that are related to the clients.

Apart from, all banking transactions are encrypted in order to be ready for the transmission through the network. The encryption process is done digitally using a signature which could be stored in smartcards or any memory medium.

The constraint linked with each cryptographic algorithm (computation resources, time to compute, key management, algorithm capability and design...) limits their use for only a given number of security services. RSA or DES for example addresses some of them like Confidentiality, Integrity, Authentication and Non-Repudiation, but never all [8]. Our new open source cryptographic algorithm should satisfy all of these security services to better secure e-banking services and enhance its protection techniques.

We want to propose a new open source encryption algorithm technique for securing the users passwords in e-banking web application. The new data encryption algorithm should protect the internet users' credentials. The new open source

application should also protect the e-banking users against SQL injection in order to increase the data security.

The work will address issues such as standardization of the new cryptosystem in internet security protocols, development of supporting infrastructure like certification authorities and implementation in servers, client devices and various applications. This work should evolve advanced secure electronic payment protocol that uses this new cryptosystem application to secure the online transactions [1].

## III. METHODOLOGY AND ENCRYPTION ALGORITHM.

The Methodology proposed in this research work is explained in this section. This section is divided into two parts, in the first part we will be looking at the modules of the system and in the second part, we will be looking at the proposed algorithm for encryption .

### A. PROPOSED MODULES

1. User Registration & Login : The user will be facilitated to register here based on his credit card details for net-banking and execute login operation based on credentials specified therein and by use of CBM analysis.
2. Machine Detection: The system will track the devices used for registration and each login executed by each user to include the following parameters MAC id, Time slot used for access, IP address of the machine as per CBM analysis.
3. OTP Authentication : After machine detection after login mechanism using CBM analysis, the system will generate validity status as valid or invalid. For invalid status the system will generate a random key for OTP verification which will transmitted on email or SMS(paid service) to ensure users authenticity.
4. E-banking features : This module will consist of banking features containing view of account details, view account balance, transfer amount to particular account no specified, mobile recharge using current balance, pay credit card bills
5. Transaction Security : This module will facilitate ensuring security of data of each transaction (account no and amount) with modified AES during storage in database.

### B. MODIFIED AES (Advanced Encryption Standard).

AES-128 makes use of 128 bit of cipher key and 128 bits of data block. It performs K rounds ( K= 10 for 128 bit AES) of encryption before finally gives the cipher text. Four operations are performed while encryption namely as *substitute byte, shift row, mix column* and *add round key*.

The key expansion unit generates ten keys of 128 bits each required by all the ten rounds. The decryption process is just the reversed method where the cipher text is fed as input to the system. After completing Nr rounds (Nr= 10 for 128 bit AES) of decryption the final plain text is generated. In the

last round the Mix-column sub-process is bypassed. The decryption process also makes use of four sub processes, namely as *inverse substitute byte*, *inverse shift row*, *inverse add round key* and *inverse mix column*. The inverse mix-column sub-process is bypassed in the last round of decryption process.

It has been found that the mix-column in encryption and inverse mix-column in decryption are most computational and hence consumes large amount of clock period. The increased clock period reduces the throughput and it has been found that mix-column and inverse mix-column operations consume the highest dynamic power consumption. Hence we are suggesting a modified algorithm for AES, in which *substitute byte*, *shift row* will remain as in the original AES while *mix column* operation is replaced by *128 permutation* operation followed by *add round key* operation. This permutation operation is similar to the one done in DES algorithm but with 128 bits instead of two 64 bit permutations. The 128 bit encryption will be done as shown in the Figure 1. The decryption will be done as shown in the Figure 2. The similarity in the encryption and decryption architecture is an added advantage while implementing on the any of the hardware platforms. The permutation box and inverse permutation box used in our algorithm are as shown in Figure 5.

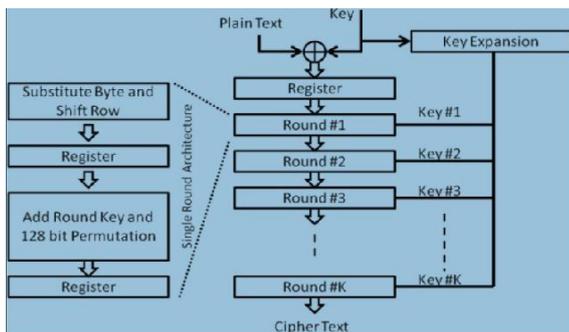


Figure 1. 128 bit AES Encryption

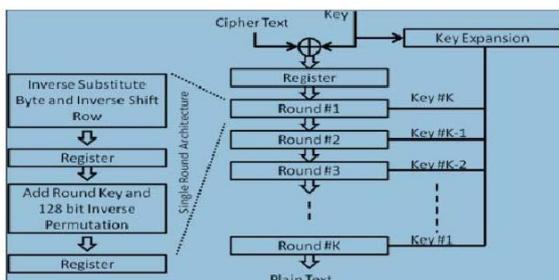


Figure 2. 128 bit AES Decryption

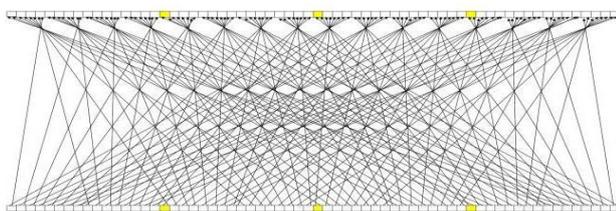


Figure 3. Permutation box implementation

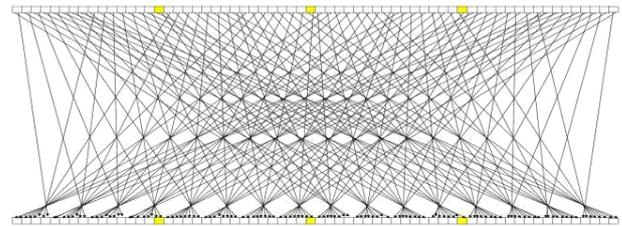


Figure 4. Inverse Permutation box implementation

114	98	82	66	50	34	18	2	116	100	84	68	52	36	20	4
118	102	86	70	54	38	22	6	120	104	88	72	56	40	24	8
122	106	90	74	58	42	26	10	124	108	92	76	60	44	28	12
126	110	94	78	62	46	30	14	128	112	96	80	64	48	32	16
113	97	81	65	49	33	17	1	115	99	83	67	51	35	19	3
117	101	85	69	53	37	21	5	119	103	87	71	55	39	23	7
121	105	89	73	57	41	25	9	123	107	91	75	59	43	27	11
125	109	93	77	61	45	29	13	127	111	95	79	63	47	31	15

(a)

72	8	80	16	88	24	96	32	104	40	112	48	120	56	128	64
71	7	79	15	87	23	95	31	103	39	111	47	119	55	127	63
70	6	78	14	86	22	94	30	102	38	110	46	118	54	126	62
69	5	77	13	85	21	93	29	101	37	109	45	117	53	125	61
68	4	76	12	84	20	92	28	100	36	108	44	116	52	124	60
67	3	75	11	83	19	91	27	99	35	107	43	115	51	123	59
66	2	74	10	82	18	90	26	98	34	106	42	114	50	122	58
65	1	73	9	81	17	89	25	97	33	105	41	113	49	121	57

(b)

Figure 5 (a) Permutation box (b) Inverse Permutation box.

In our security algorithm, except the replacement of Mix-column with permutation operation and replacement of Inverse Mix-column with Inverse Permutation box as shown in Figure 4 and 5, where the data scrambling is between 128 bits rather than 64 bits, which increases the more resistance to attacks compared to 64 bit permutation boxes. The rest of the encryption and decryption process would remain similar to the original 128 bit AES, as shown in Figure 1 and Figure 2. Our modified algorithm has 10 rounds of key expansion and 10 rounds of encryption. The decryption process requires the round key in the reverse order to that of encryption. The same round keys generated while encryption are used for decryption.

The implementation environment was as follows:

- 1) The Permutation box is of 128 bit instead of two 64 bit permutations. This reduced the computation time further and the inter bit delay differences achieved are very less.
- 2) The implementation was on a dedicated hardware designed for the algorithm on 180nm CMOS technology. Therefore, the over heads of the processor and OS performance were taken into consideration. The original AES implementation was done by authors on the same FPGA platform [17]. The highest throughput achieved was 1Gbps using rolled architecture of 128 bit data bus. Our implementation of modified AES using the rolled

architecture of 128 bit data bus achieved a throughput of 2.087Gbps on Xilinx Vertex4 FPGA. The only drawback of our method is larger memory used for storing the 128bit permutation box.

The implementation results for our design are:

- The design was successfully clocked at 163MHz.
- Total gates required for encryption and decryption were: 107K gates (including memory in equivalent gate terms).
- The throughput is  $163\text{MHz} \times 128/10 = 2.087\text{Gbps}$ .
- Power consumption is 23.84mw at 163Mz.
- The scrambling of data bits with our permutation box has resulted into less inter-path delay differences.

We achieved a throughput of 2.087 Gbps which means 16.305M, 128 bit blocks per sec. OR 61.332ns for each block of 128 bit data. We have tested on Vertex 4 FPGA platform. The most important contribution of our implementation is that the inter bit delay differences are very less. The maximum delay between the most critical path and shortest data path from input to output is found to be 1.0118ns. This result has helped us in reducing the clock period, as the data arrival uncertainty was reduced. The table below summarizes the comparison among different AES designs [1].

#### IV. CONFIDENTIAL BUILDING METRIC (CBM).

As mentioned in earlier sections, every account holder (client) registered and identified by the Bank will be able to use online banking and transaction services. Every transaction done by the client with the bank is encrypted using 128 bit modified AES described above. The primary concern in this part of the design is the key distribution. The key has to be renewed every time whenever the bank asks to change the access and transaction password of the client. This is normal practice with the banks to request or make it mandatory to the client to change the access and transaction passwords regularly. The passwords are expired after every certain period is passed. We additionally suggest another level of security using Confidence Building Metric (CBM). Such that, in order to increase the protection by still getting faster computation, we may use the concept of Confidence Building Metrics (CBMs) based on the certain parameters of access to the bank portal by the account holder. The CBM is mapped on a scale of 0 to 10. The value of CBM increments with the following list of events as mentioned below:

- MAC id of the computer regularly used
- Time slot in day or night whichever is used often or regularly
- IP address of the machine regularly used for transaction
- Use of virtual keyboard used every time the transactions made
- Amount of transaction within a certain limit pre-declared by the client

Each of these parameters or events mentioned above increments the CBM by certain value. The distribution of these values can be a design consideration of the bank.

Similarly the CBM also decrements whenever certain events do occur as mentioned below:

- New MAC id used other than the regularly one
- Transactions being done at odd hours others than the regular ones
- Different IP address of the machine or device other than the regular one
- Virtual keyboard not used
- Amount of transaction exceeds the pre-declared limit. New access and transaction password set.

The banks will maintain a set of security questions already configured or registered by the client. The banks may make the registration of security questions and their answers mandatory. Whenever the CBM decrements than the previous value an additionally security question is asked to the client from the set on random basis. And whenever the CBM increments no security questions are asked and the transaction can be done based on the access and transaction passwords while encryption is compulsorily done for every data transfer. This makes a three tier system of verification and validation of the genuine client requesting the transaction. This three tier system is as shown in Figure 6.

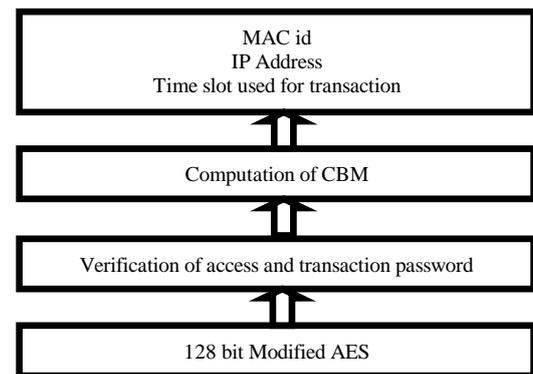


Figure 6. Tier structure of security system

The advantage of the above system is that it can be implemented on any platform at the client side including even the mobile phones with any OS [1].

#### V. OTP GENERATION.

An OTP is a generated password which only valid once. The user is given a device that can generate an OTP using an algorithm and cryptographic keys. On the server side, an authentication server can check the validity of the password by sharing the same algorithm and keys. Several software or devices can be used to generate the OTP, for example personal digital assistants, mobile phones, dedicated hardware tokens as it the most secure smart cards is devices among all the OTP generator provide tamper-resistant two-factor authentication: a PIN to unlock the OTP generator (something you know), and the OTP smart card itself (something you have). Figure 7 illustrates the three steps that required to generate an OTP: the collection of some external data, such as the time for synchronous OTP or a challenge for an asynchronous OTP, a

ciphering algorithm with secret keys shared by the device and the authentication server, and finally a formatting step that sets the size of the OTP to typically six to eight digits[8].

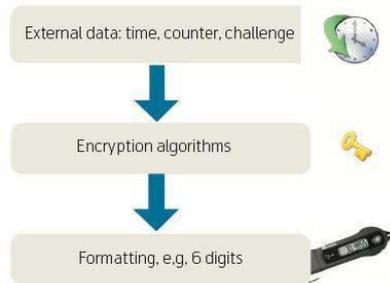


Figure 7. The generation of One-Time passwords

Until recently, OTP solutions were based on proprietary and often patented time-based or event-based algorithms. In 2005, OATH-HOTP [16] was defined as an open standard by major actors in the industry. This open standard allows multi-sourcing of the OTP generating devices and authentication servers from different vendors. The HOTP algorithm is based on a secret key and a counter shared by the device and the server, and uses standard algorithms such as SHA-1 and HMAC.

OTP has carried more advantages over PKI as it does not require the deployment of smart card readers, drivers and PC software. However in terms of features, OTP only provides identification and authentication, whereas PKI provides addition encryption and signature. OTP being a password-based authentication is also vulnerable to man-in-the-middle attacks, such as phishing scams. Since there is no mutual authentication of the PC and the internet service provider server, an attacker can intercept an OTP using a mock-up site, and impersonate the user to the real internet web site.

The proposed authentication system performed the user authentication and digital signatures using authorized certificates in the same way as the existing authentication. To recognize and convert the code, we generate the mobile OTP code into a two-dimensional barcode using user's transfer information (TI), requested transfer time (T) and the hashed serial number (SN) of user's mobile device instead of security card. The authentication process of proposed system is shown below the Fig. 8.



Figure 8. A propose Authentication System

- When user execute the generated OTP, mobile device generate the OTP by reads the transfer information (TI), perceived value of time (T) and hashed serial number (SN) of user's mobile device are shared with the certification authority (CA). And output the generated OTP on the screen of mobile devices.
- User input the generated OTP code from mobile device on the screen.
- Server (Bank) sent OTP to certification authority (CA) to received OTP from user.
- Certification authority (CA) compared by received OTP code (OTP1) and generated the OTP code (OTP2), sent to server (Bank) to for OTP code approval.
- When the server (Bank) received approve of OTP from certification authority (CA), it will verify the entered OTP code with user consistent value and user digital signature. If the approve of OTP value does not receive, the transfer will be canceled.
- Authorized user signed his certificates to complete the transfer.
- Server (Bank) to verify the digital signature and final approve of transfer.

## VI. CONCLUSION AND FUTURE WORK.

E-banking is a form of banking where money is transferred through an exchange of electronic signals between financial institutions. The security of data record transaction has brought many concerns from different perspectives: government, businesses, banks, individuals and technology. Financial institutions are achieving the security of e-Banking

data record transaction by methods of cryptography, which deals with encryption of data. Here we have proposed a new encryption algorithm that is based on AES using open source symmetric key encryption algorithm. This modified AES algorithm provides better security for the e-banking services and overcomes the problem of computational overhead by reducing the calculation time of the algorithm. Comparative study with traditional encryption algorithms is shown the superiority of the modified algorithm. A new innovated E-Banking Security Tier using Confidence Building Metric (CBM) and Modified AES was presented to be another level of protection. The CBMs are computed based on certain parameters and can be implemented on any platform at the client side. Some improvements on the deployment of our modified AES will be considered as a future work taking into consideration on the importance level of each e-banking transaction record.

#### REFERENCES

- [1] Adel Khalifi, Maher Aburrous, Manar Abu Talib, P. V. S. Shastry, "Enhancing Protection Techniques of E-Banking Security Services Using Open Source Cryptographic Algorithms", 14<sup>th</sup> ACIS International Conferences on Software Engineering, Artificial Intelligences, Networking and Parallel / Distributed Computing, 2013.
- [2] Jorge Aguila Vila, Jetzabel Serna-Olvera & Luis Fernandez, Manel Medina & Andreas Sfakianakis, "A Professional View on eBanking Authentication: Challenges and Recommendations", 9<sup>th</sup> International Conference on Information Assurance and Security (IAS), 2013.
- [3] J. Choubey and B. Choubey, "Secure User Authentication in Internet Banking: A Qualitative Survey", International Journal of Innovation, Management and Technology, vol. 4, no. 2, 2013.
- [4] D. B. Eran Kalige, "A Case Study of Eurograbber: How 36 Million Euros was Stolen via Malware", Versafe (White paper), 2012.
- [5] D. Marcus and R. Sherstobitoff, "Dissecting Operation High Roller", McAfee, 2012.
- [6] D. B. Eran Kalige, "A Case Study of Eurograbber: How 36 Million Euros was Stolen via Malware", Versafe (White paper), 2012.
- [7] L. Peotta, M. D. Holtz, B. M. David, F. G. Deus and R. T. d. S. Jr, "A Formal Classification of Internet Banking Attacks and Vulnerabilities", International Journal of Computer Science & Information Technology, pp. 186-196, 2011.
- [8] Kamrul Hasan E-Banking in Bangladesh : The Future of Banking, in Proceedings of Annual Paris Conference on Money Economy and Management Annual Paris Conference on Money Economy and Management (2011).
- [9] A. Fatima, "E-Banking Security Issues Is There A Solution in Biometrics?", Journal of Internet Banking and Commerce, August 2011, vol. 16, no. 2, 2011.
- [10] R. Chouhan and V. S. Rathore, "E-Banking Security and Authentication Issues", International Referred Research Journal, 2011.
- [11] Young Sil Lee, Nack Kim, Hyotack Lim, Heringkuk Jo, Hoon Jae Lee, "Online Banking Authentication System using Mobile OTP with QR-Code.", National Research Foundation Project and Dongseo Frontier Project, 2010.
- [12] M. R. Nami, "E-Banking: Issues and Challenges", in 10th A CIS International Conference on Software Engineering, Artificial Intelligences, Networking and Parallel/ Distributed Computing, 2009.
- [13] C. K. Dimitriadis, "Analyzing the Security of Internet Banking Authentication Mechanisms", Information Systems Control Journal, 2007.
- [14] C. K. Dimitriadis, "Analyzing the Security of Internet Bank- ing Authentication Mechanisms", Information Systems Control Journal, 2007.
- [15] A. Hiltgen, T. Krampan and T. Weigold, "Secure Internet Banking Authentication", IEEE Security & Privacy, 2006.
- [16] IETF RFC 4226, HOTP: An HMAC-Based One-Time Password Algorithm, Dec. 2005.
- [17] Joseph Zambreno, David Nguyen, Alok Choudhary, "Exploring area/delay tradeoffs in an AES FPGA implementation", FPL 2004 proceedings, LNCS vol 3203, pp.575-585.

#### **JELPHINA LOBO**

M.E.I.T., Thakur College of Engineering and Technology,  
University of Mumbai, India.

#### **ZAHIR AALAM**

Associate Professor, TPO, TCET, Mumbai, India.