E-Voting through Biometrics and Cryptography-Steganography Technique with conjunction of GSM Modem

Shobha lokhande Dipali sawant Nazneen Sayyad Mamata Yengul

D.D.Pukale
Computer Engineering Department of BVCOEW
Pune University,Pune 411 043,India

ABSTRACT

Biometric as well as Password security to voter accounts is provided using Cryptography and Stenography. For implementing this, cover image is used for Stenography and key is used for Cryptography. The basic concept is to merge the secret key with the cover image on the basis of key image which results a stego image, which looks quite similar to the cover image but not detectable by human eye. The key image is a Biometric measure, such as a fingerprint image. As the hackers have to find both secret key and the template so proper use of Cryptography provides security to the system. The main purpose of system is to provide proper authentication of voter for voting system.

Keywords

OnlineVoting,Steganography, Biometric, Cryptography

1. INTRODUCTION

Elections allows people to choose their representatives and express their preferences for how they will be governed naturally, the integrity of the election process is fundamental to the integrity of democracy itself. [4]

The election system must be -

1. Sufficiently robust to withstand a variety of fraudulent behaviors.

2.Sufficiently transparent and comprehensible that voters and candidates can accept the results of an election. [5]

The design of a “good” voting system, whether electronic or using traditional paper ballots or mechanical devices , must satisfy a number of sometimes competing criteria.

1. Anonymity - To guarantee the voters safety when voting against a malevolent candidate and to guarantee that voter have no proof that proves which candidates received their votes.

2. Tamper-resistant: The voting system must also be tamper-proof to thwart a wide range of attacks, including ballot stuffing by voters and incorrect tallying by insiders.

3. Human factors: A voting system must be comprehensible and usable by the entire voting population, regardless of age, infirmity or disability.

Other requirements of such a system could be viewed as scalability, speed and accuracy and authentication. By Authentication, once we are sure that a voter is genuine , we can easily address other issues like anonymity and tamper resistance.[4] Some of the existing solutions of computerized voting systems are explained by Armen and Morelli and highlighted their vulnerabilities. They include Punch Card Systems, Global Election Management System (GEMS) and Direct Recording Electronic (DRE). [1]

As these systems are stand alone systems, they lack in ability of voting from anywhere. That is why the actual notion of online voting is missing in those systems. Rest of the paper is organized as follows. In the next section basic methodology is explained in subsections namely cover image creation, secret key expansion using hashing, Embedding algorithm, authentication algorithm and voter account maintenance. Analysis is done in section 3. Finally, we conclude in the last section.

2. PROPOSED SYSYTEM

Following figures shows the whole architecture of

system.

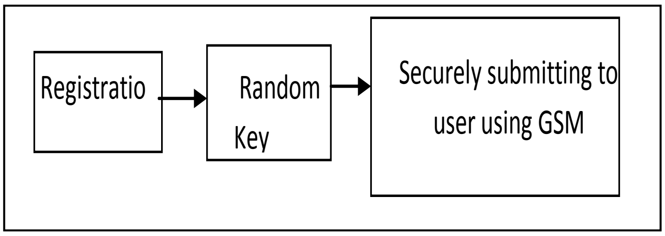

Figure 2.1 Voting Server Architecher

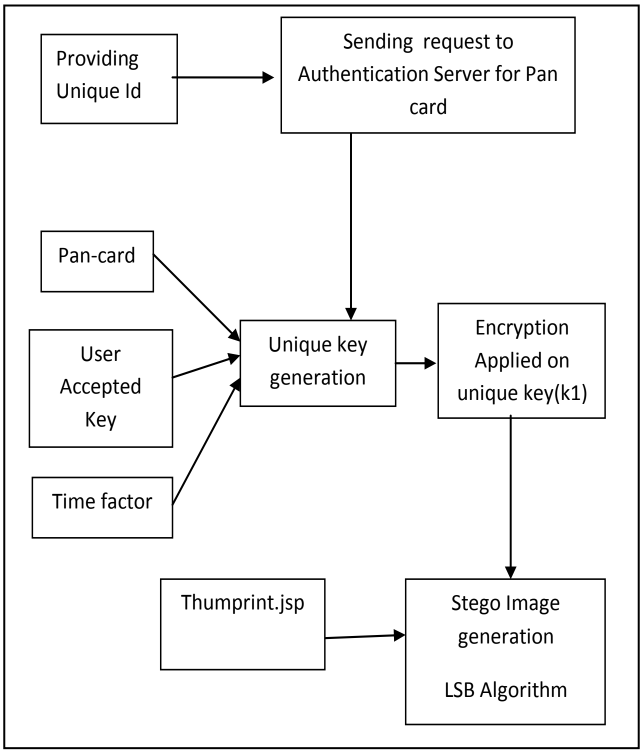

Figure 2.2 Voting Architecture

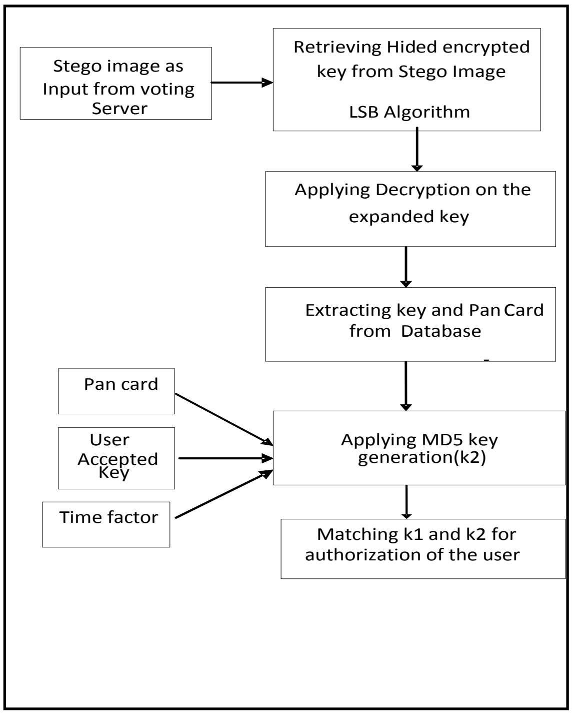
Stego image as Input from voting Server
Retrieving Hided encrypted key from Stego Image
LSB Algorithm
Applying Decryption on the expanded key
Extracting key and Pan Card from Database
Pan card
User Accepted Key
Time factor
Applying MD5 key generation(k2)
Matching k1 and k2 for authorization of the user

Figure 2.3 Authentication Server Architecture

In Registration module the user will be provided with a random key of 4 characters that key will made unique by adding his unique data like his pan-card number.

Our System would consist of two Server one would be Voting Server and one would be Authentication Server. Authentication Server would have a legal access to the centralized voter database. The Voter will provide his voting to voting Server The Input provide by voter to voting server would include his Unique Id (Adhar Card or any other),key and his thumb print.

Users Unique card will be used for requesting his pan-card from authentication Server after receiving Pan-card, voting server will merge the Pan card details, Key provided and current time factor and apply MD5 Algorithm to generate a key called k1 .this generated key is hided in accepted users thumb print image for generating stego image. After this the stego image is send to Authentication Server. Authentication Server removes the hided key naming it k1 .Then Authentication Server extracts the corresponding users key and pan card from the centralized database , current time factor and applies MD5 Algorithm to generate a new key k2.So the two keys k1 and k2 are matched for authorization purpose.

2.1 Administrator Login

Let the admin user name be x and password be **y** Let the Validation process performed on username and password be **V** The Result is R if username x and password y is correct according to the database

Accept(**x**,**y**) => (username,password)

If V(**x**,**y**) => Valid => **R**

Voter Voting

Let the voters voter ID be **v**, key be **k** and Pan Card be **p ,**

The voter id is **v** taken from user by voting server and submitted to the Authentication server by process say Accept.

Accept (**v**) => voterID => Authentication.

Accordingly the Authentication Server uses this voter id of the voter and fetches his pancard **p** By process so called fetch

Authentication => Fetch (**p**) => Voting Server

From user key **k** is taken by process called getKey.

getKey(**k**) => User

Now MD5 is applied on key **k**, pancard **p** and Time **t** and unique key of 128 bits is created

MD5(**k**,**p**,**t**) => Unique key(k1)(128 bits)

The above created 128 bits unique key is hidden in one image to get stego image **s** by process called createstego.

Createstego(**k1**,**i**) => stego image(**s**)

The created stego image is send to the Authentication server for authentication purpose.To get the new key **k1** by process called authkey

Authkey(**s**) => **k1**

Independedntly at the authentication server is applied on key k1, pancard p1 and Time t1 and unique key of 128 bits is created

MD5(**k**,**p**,**t**) => Unique key(**k2**)(128 bits)

Now finally both calculated key **k2** and received key **k1** is match if they are same then the user Is valid

k1 == **k2** => voter is valid

There are some pre-requisites to support such a system. Firstly, each and every individual in the country should be provided with a Personal Identification Number, such as SSN (Social Security numbers) in some countries. This is needed for maintenance of voter accounts in the database.

Secondly, we need Thumb Impressions (fingerprint images) of all the individuals. Thirdly, during the account creation every individual will be provided with a system generated. Secret key which he/she should not disclose to anybody. This will be needed to cast the vote. Assuming all voter's information in a country is securely collected, biometric reader available for voting, the system is online during the election period only, the methodology is as follows. To cast a vote, a voter logs in to the system by entering the personal identification number and secret key. Along with this voter has to give the thumb impression on the fingerprint sensor. The system will generate the cover image and embed the secret key into it according to the predefined procedure to generate the stego image. Now this stego image will be sent securely to the server for voter authentication. Fingerprint forgery may be restricted by using advanced fingerprint readers which employ Ultrasonic and Capacitance. At the server side, it will use the Optical Character Recognition technique to read the personal identification number represented on the image. After reading it, the server will find out the details of that individual from the database. These details will be his/her fingerprint image and secret key.

Using these details, the image can be decoded to find out the embedded message which should be the secret key of that individual. Once authentication is complete, the voter will be allowed to vote. In this next page, all the details regarding the voting boundaries of that individual will be shown. Here voter can select the desired candidate and finalize the vote. After casting the vote, the account will be closed and in the database the voted bit will be set to one for that voter. Figure2.4 shows the basic mechanism:

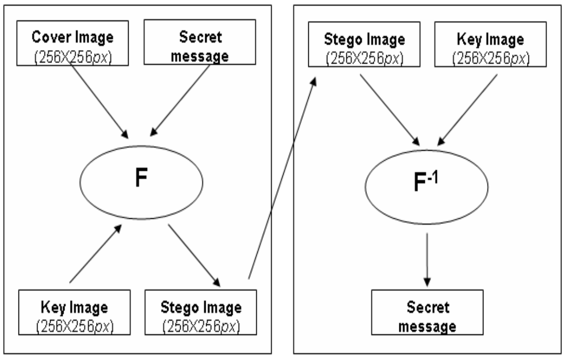

Figure2.4

Now we will introduce all the steps to be performed for logging into a voter account in a little more detail.[1]

2.2. Cover Image Creation

Every voter should have a 16-digit personal identification number. This number will be automatically written over a base image in predefined font style & size. Let us use 256*256 pixels bitmap cover image. The base image should be clear so that the text written over it is machine readable. This image will be finally modified into a stego image and sent over insecure channel. The base image is a default image for the system, same for all. Cover image is a simple inscription of personal identification number over the base image. So, the cover image for every voter will be same except the digits written over it shown in the Figure 2.5:

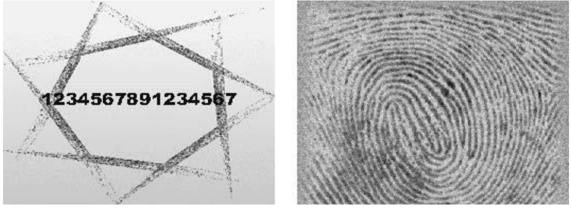

Figure 2.5

2.3. Secret Key Expansion Using Hashing

The secret key plays very important role in the whole process. It should not be compromised in any case. There is a limitation with the secret key here, as the system is designed for general public which is quite negligent in these issues, we cant keep the key too long. It should be short enough to be remembered by everybody. For explanation purpose we are assuming it to be a 4-digit number, similar to ATM PIN. This 4-digit PIN can easily be represented using 2 bytes.

But 2 byte data looks very much vulnerable in terms of length. As we have to finally embed it into the image, which is quite big. The cover image is a 24-bit image where every pixel is represented using three bytes. So, we have 3 ∗ 2^16 byte data in total. Now hiding only 2 bytes in this much space will not fully exploit the resources in terms of cryptography. This is because the algorithm we are using provides both cryptography and steganography at the same time. Steganography says its good as the statistical properties of the cover image will remain intact due to under performed modification [1]. The eavesdroppers will never be able to deduce that some data is hidden in the image. But if somehow they know that it is a stego image, they can easily extract the PIN From the cryptography point of view, the key image under utilized as well. As the fingerprint image is of the same dimension, we will be exploiting very less features of the key image. So, to increase the complexity of analysis, the 2 byte secret key is expanded to 32 byte key by applying MD5 hashing algorithm [2]. Now these 160 bits will become a part of the actual secret message. When the secret message is embedded in the cover image, its statistical properties will not remain same. The stego image will remain more complex to be analyzed because more features of the key image are utilized in this case. So, even if eavesdroppers know that this is a stego image, it would be more difficult for them to predict the embedded data.

2.4. Generation of the secret message

In this phase of the methodology, we will get a 160 bit secret message from a 16 bit secret key. Firstly, the secret key is concatenated with the time-stamp value. The timestamp is a 32 bit value which represents the current date. Now we will apply MD5 algorithm to get a 128 bit hash code for that key. Now the same time-stamp is concatenated with this hash code to get the secret message. So, our secret message will be of

160 bit length. As the actual secret key is never embedded in the stego image, there will be no chance of predicting secret key from it. The mechanism is shown in Figure 2.6:

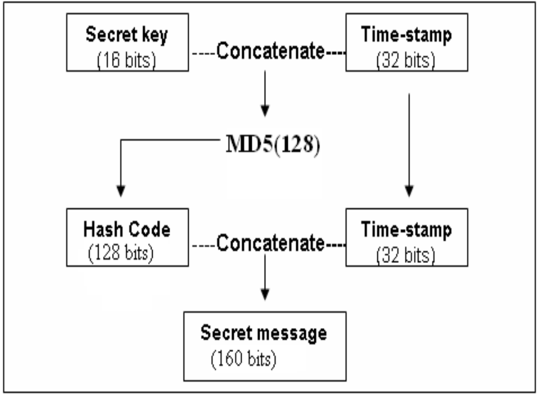

Figure2.6

2.5. Description of Embedding Algorithm

The embedding algorithm makes use of a stegocryptographic model. The model easily unifies cryptographic and steganographic models. It basically results as a steganographic one with the addition of a new element as the key image. It finally delivers cryptographic functionality while preserving its steganographic nature.The output of this embedding process is a stego image S and the inputs are expanded secret key concatenated with time-stamp, i.e. secret message, a cover image and the key image.

Module : implementing Steganography

Usually 24-bit or 8-bit files are used to store digital images. The former one provides more space for information hiding; however, it can be quite large. The colored representations of the pixels are derived from three primary colors: red, green and blue. 24-bit images use 3 bytes for each pixel, where each primary color is represented by 1 byte. Using 24-bit images each pixel can represent 16,777,216 color values. We can use the lower two bits of these color channels to hide data, then the maximum color change in a pixel could be of 64-color values, but this causes so little change that is undetectable for the human vision system. This simple method is known as Least Significant Bit insertion as in figure 2.7 Algorithm to embed the encrypted data:

This algorithm is only for embedding a character (8-bit). For embedding the entire message, the steps in the algorithm are repeated. The output obtained as a result of encryption performed in Module 3 is embedded in an image which is of Portable Network Graphics format i.e. image with '.png' extension. The process of embedding consists of the following steps:

Step 1: The image is selected initially, in which data has to be embedded.
Step 2: The total number of pixels in the image is calculated by using the formula 'width x height'.
Step 3: The color intensities of each and every pixel is retrieved and stored in an array. Each pixel constitutes of 3 bytes, where each byte represents one of the three primary colors i.e. RGB.
Step 4: AND operation is performed on each byte of the pixel along with the binary equivalent of 252. The result obtained is the byte value with the last two bits as '00'.

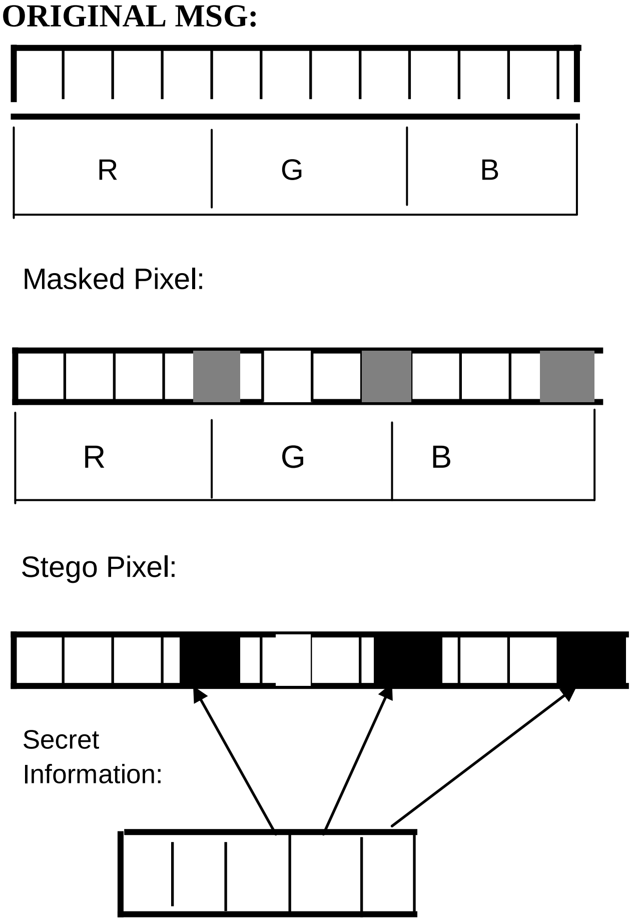

Figure 2.7

Step 5: The cipher text is AND operated with the binary equivalent of '03' to retrieve the last two bits of the message.
Step 6: The OR operation is performed with the output of step 4 and step 5.
Step 7: The output of step 6 becomes the new intensity of the Red color. For Green and Blue color step 4 is repeated and before doing step 5 right bit shifting is performed to the cipher text in the incremental order of 2 till all the 8 bits are embedded.

To retrieve the cipher text from the image, the reverse steps of the algorithm mentioned above is to be performed.

2.6. Voter Account Maintenance

Once any individual passes the authenticity criteria, he/she will be logged into his/her voting account. We can easily restrict a voter from logging into his/her voting account more than once during elections. Once a particular voter is authenticated by the system, a secure channel will be established using https and then he/she will be able to cast the vote. The vote will remain secret in every sense, i.e., it will not be reflected anywhere in the database that which user has voted for whom. Finally, the account will be closed and that user will not be able to log back in by any means again. This completes the voting process. The authentication mechanism makes use of both, biometric measures as well as secret key. If any of these properties are tempered by any individual, it can be easily detected and the request will be rejected from the server side.

3. ANALYSIS

In this section we analyze the performance of our algorithm with respect to both cryptography and steganography.

A. Cryptographic Performance

The embedding process consists of three major portions. These are hashing, the pseudo random function and one time pad. For hashing we are using MD5, which gives us 128 bit hash code. This hash code depends upon secret key and the time-stamp value making it different for every individual every time he/she votes. Also, if someone gets the required bits out of the stego image, secret key could still can not be predicted because of hashing. The only concern with MD5 is its speed. Any delay is undesirable in this system as many people will be voting all around the country. Also, we have to use hashing on both ends for each and every request which adds to our concerns. If somehow it is compromised, the attacker can fetch the voter key by trying all possible keys. So, the pseudo random function should be implemented with utmost security. Third and the most important step is the actual embedding of bits. The mechanism is similar to one time padding technique which is theoretically unbreakable. If a cryptanalyst has a cipher text string encrypted using a random key which has been used only once, the cryptanalyst can do no better than to guess the plain text of the same length. One more advantage of this technique is the speed. It can be performed very quickly. The algorithm is found to be strong if key is never reused. In our case, the key image never changes but the data we are embedding changes every time. Also, the authentication algorithm needs the hash code, which can not be replaced successfully until the attacker knows the secret key. So, collectively we can say that the cryptographic performance of this algorithm is good.

3.1 Steganographic Performance

The aim of steganography is to hide information imperceptibly into a cover, so that the presence of hidden data cannot be diagnosed. Here we have used LSB steganography, in which the lowest bit plane of a bitmap image is used to convey the secret data. Because the eye cannot detect the very small perturbations it introduces into an image and simple to implement. In this algorithm, the secret message is fewer bits in length than the number of pixels in the cover image also the pseudo random permutation ensures that changes are spread uniformly throughout the image. More is the number of bits modified; more will be the change in the statistical properties of any image[1]. Here we are modifying only 0.017 percent of bits available in the cover image making it difficult to be detected as steganographic one. So, from the steganographic perspective, the statistical properties of the cover image are least hurt, hence resulting in better security.

4. CONCLUSION

In this paper we have presented a method for integrating cryptography and steganography. The strength of our system resides in the new concept of key image. We are also able to change the cover coefficients randomly. This strategy does not give any chance to steganalytic tools of searching for a predictable set of modifications. Also, considering the complexity of elections, we have provided sufficient proof of authenticity of an individual in form of both biometric measures and secret key. As future work, we will be trying to improve two considerable aspects of the algorithm, namely, speed and dependence on pseudo random function.

5. ACKNOWLEDGMENTS

We express our sincere thanks to our project guide **Prof. D. D. PUKALE** who always being with presence & constant, constructive criticism to made project successful.

I would also like to thank all the staff of **COMPUTER DEPARTMENT** for their valuable guidance, suggestion and support through the project work, who has given co-operation for the project with personal attention.

We again take it as great privilege to express our heartfelt thanks to our principal and Head of Department for their valuable suggestion for developing project at every state.

Above all we express our deepest gratitude to all of them for their kind-hearted support which helped us a lot during project development. They offered us plenty of opportunities while working with them, rendered us in valuable help & helped us linking practical knowledge with theoretical one taught to us in our college.

At the last we are thankful to our friends, colleagues for the inspirational help for they gave us through a project work.

6. REFERENCES

[1] Shivendra Katiyar, Kullai Reddy Meka, Ferdous A. Barbhuiya, Sukumar Nandi,*"Online Voting System Powered By Biometric Security Using Steganography",* 2011 Second International Conference on Emerging Applications of Information Technology

[2] William Stallings, *"Cryptography and Network Security, Principles and Practices"*, Third Edition, pp. 67-68 and 317-375,Prentice Hall, 2003.

[3] Bruce Schneier, *"Applied Cryptography"*, Second Edition: Protocols, Algorithms, and Source Code in C, John Wiley and Sons, 1996.

[4] Tadayoshi Kohno, Adam Stubblefield, Aviel D. Rubin, and Dan S. Wallach, *"Analysis of an Electronic Voting System"*, Proc.IEEE Symposium on Security and Privacy (May, 2004), found at http://avirubin.com/vote/analysis/index.html

[5] Robert Krimmer (Ed.) *,"Electronic Voting 2006"*2nd International Workshop Co-organized by Council of Europe, ESF TED, IFIP WG 8.5 and E-Voting.CC.