International Journal of Computer Science and Informatics

Volume 3 | Issue 1 Article 7

July 2013

A Novel Steganographic Algorithm and Hashing to Improve Authentication using Mobile Phones

G. Sumalatha
Dept. of IT, Sree Nidhi Institute of Science & Technology, sumalatha_adb@yahoo.com

P. Madhuravani
Dept. of CSE Malla Reddy College of Engineering & Technology, madhuravani_b@yahoo.com

Recommended Citation

Sumalatha, G. and Madhuravani, P. (2013) "A Novel Steganographic Algorithm and Hashing to Improve Authentication using Mobile Phones," *International Journal of Computer Science and Informatics*: Vol. 3 : Iss. 1 , Article 7.
DOI: 10.47893/IJCSI.2013.1115
Available at: https://www.interscience.in/ijcsi/vol3/iss1/7

A Novel Steganographic Algorithm and Hashing to Improve Authentication using Mobile Phones

G. Sumalatha[1] & P. Madhuravani[2]

[1]Dept. of IT, Sree Nidhi Institute of Science & Technology
[2]Dept. of CSE Malla Reddy College of Engineering & Technology
E-mail : sumalatha_adb@yahoo.com[1], madhuravani_b@yahoo.com [2]

***Abstract* -** Security has become major issue in all online services. Strong authentication is provided conveniently using mobile phone as security token. While sending security token to user mobile via GSM network, it is vulnerable to several attacks. We can avoid them using Steganography which enforces the security of hashing algorithms. We can share tokens securely with steganography and authentication to prevent participants from the intentional provision of a false stego-image. However, an illegitimate person can easily manipulate the stego-image for successful authentication but cannot recover the secret image. We propose a new authentication method called Σ-hash, which combines an efficient steganographic algorithm and hashing. This paper tells how Σ-Hash can be used for securing OTP(One Time Password) from tampering attacks and other applications.

***Key words* -** *OTP,Stegnography, stego-image and ∑-hash.*

I. INTRODUCTION

As the popularity of the Internet increases the number of frauds and abuses is literally exploding. Most serious is the theft of identity which causes grave damages both for the victim and also his entourage such as employee, banks, hobby clubs, etc. The protection of digital identities is getting more and more crucial. Strong authentication solutions require often two identification factors i.e, in addition to the first factor "something you know" represented by passwords it is introduced a second factor "something you have" materialized by a security token. The introduction of the additional device could be costly for the service providers in terms of deployment and administration at the same time as it could be inconvenient for mobile users. Furthermore, there is very little re-use or sharing such that the same security token can be used for several systems. To remedy the situation, there are proposed several authentication solutions that avoid introducing extra device by re-using existing devices, namely the mobile phone or the SIM cards.

This paper presents a study of the multi factor authentication solutions using mobile phone as security token and the use of steganography and tells how it avoids attacks to hash functions.

The paper starts with the explanation of multi-factor authentication in Section II. Section III describes the architecture of the strong authentication using mobile phones. How OTP is generated and send to users is decribed in Section IV. Section V describes how the steganography and hash functions are combined to improve the authentication. Section VI describes the new steganograpy method. Section VII describes that no collision attacks to proposed scheme and finally section VIII concludes the paper.

II. MULTI-FACTOR AUTHENTICATION

Authentication confirms the identity of the person. Once the identity of the user or system is validated, access is granted.

US Federal regulators consistently recognize three authentication factors:

"Existing authentication methodologies involve three basic “factors”:

- **Something you know:**
  - Passwords
- **Something you have:**
  - One-time password tokens

  - ATM cards
- **Something you are:**
  - Biometrics

Multifactor authentication (MFA) is a security system in which more than one form of authentication is implemented to verify the legitimacy of a transaction. In contrast, single factor authentication (SFA) involves only a user ID and password. In two-factor authentication, the user provides dual means of identification, one of which is typically a physical token, such as a card, and the other of which is typically something memorized, such as a security code. Additional authentication methods that can be used in MFA include verification such as finger scanning, iris recognition, facial recognition and voice ID. In addition to these methods, smart cards and other electronic devices can be used along with the traditional user ID and password.

We propose a system that uses mobile phones to generate tokens that will save the organizations in terms of money and maintenance.

III. ARCHITECTURE OF THE STRONG AUTHENTICATION USING MOBILE PHONE

As shown in the figure 1 the user must have access to a computer connected to the Internet and be in possession of a mobile phone with an operating SIM card. If the computer and mobile phone is equipped with Bluetooth higher usability can be obtained. Through the Internet browser the user can access web services provided by service providers. The service provider (SP) is connected to an Authentication Server (AS) that will handle the authentication on behalf of the SP. The AS is connected to the GSM network which enables it to communicate with the user's mobile phone and the operators Authentication Center (AuC). The AS is composed of two parts, an authenticator and an AAA (Authentication, Authorization and Accounting) server. The authenticator communicates with the client and relays messages to the AAA server which handles the authentication.

In an authentication scheme which uses two separate devices that communicate over two different networks it is very important to ensure that it is the same user that controls both devices.

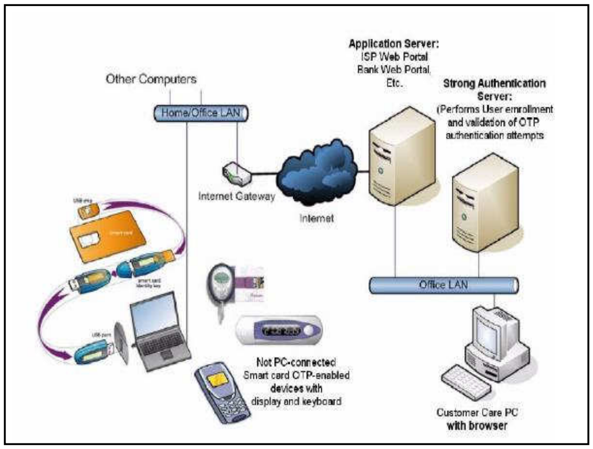

Application Server:
ISP Web Portal
Bank Web Portal
Etc.
Other Computers
Strong Authentication Server:
(Performs User enrollment and validation of OTP authentication attempts
Home/Office LAN
Internet Gateway
Internet
Office LAN
Not PC-connected Smart card OTP-enabled devices with display and keyboard
Customer Care PC with browser

Fig. 1 : Authentication in Mobile phones

IV. OTP GENERATION

The OTP Authentication System provides strict access control, based on strong two factor authentication.

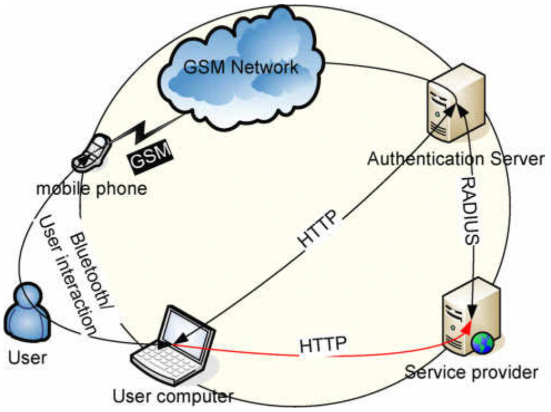


Fig. 2 : OTP Authentication System

In order to secure the system, the generated OTP must be hard to guess, retrieve, or trace by hackers. Therefore, it’s very important to develop a secure OTP generating algorithm. Several factors can be used by the OTP algorithm to generate a difficult-to-guess password. OTP algorithms takes mobile identification factors as input and are concatenated and the result is hashed using any hashing technique. The message is then shrunk to an administrator-specified length by breaking it into two halves and XOR-ing the two halves repeatedly.

Even though hash functions are carefully designed to satisfy the required security properties, they are still vulnerable to collision attacks. Due to their nature, it will always be possible to find two different inputs that will produce the same output. Using the birthday paradox [14] an attacker can find a collision for a hash function of range r in $r^{1/2}$ operations. This is considered the simplest attacking method, equivalent to a brute force attack.

Even though these attacks are only theoretical, they demonstrate that the currently most widely used hash algorithms are indeed vulnerable and eventually, faster and more practical attacks will be discovered. Actually, in the case of MD5, the authors of [13] managed to create two X.509 certificates with different public keys but the same MD5 hash, showing that a practical attack is feasible.

V. THE Σ-HASH SCHEME

The proposed scheme combines an efficient steganography algorithm and hash functions in order to improve the authentication.

Hashing :

The Σ-Hash scheme [9] involves three steps: hashing, embedding and Σ-Hashing, which corresponds to hashing the stego-object. Let *M* denote the original message that will be Σ-Hashed. During the first step *M* is hashed using any hashing algorithm f_h, to produce the hash value *H*:

$f_h(M) = H$

In the second step we embed the hash value *H* to *M*. This can be done by using an efficient and simple steganographic algorithm called Selected LSB denoted by f_s. The stego-key for the embedding process will be again the hash value *H* that was produced in the first step. The output of this step will be a stego-object called M_S as follows:

$f_s(M, H, H) = M_S$

By choosing *H* as a key we eliminate the need for a key exchange and maintenance, as the hash value will be exchanged anyway. Furthermore, the steganographic process ensures that the secret message, in our case the hash value, will be spread across the original message, regardless of its size and without affecting its appearance and functionality. Thus, the original object will remain functional, regardless of the embedded message.

In the third step the stego-object M_S is hashed using any hashing algorithm, possibly the same as in the first step:

$f_h(M_S) = H_S$

where H_S is the hash value of the stego-object.

We have now computed two different hash values for seemingly the same object. The first one, *H*, is the usual hash value, while the second one, H_S, is computed over an alternate version of the original object, which contains a secret message, embedded to it using Selected LSB steganographic algorithm. The final hash function that will be used is produced by XOR-ing *H* and H_S:

$\Sigma\text{-}Hash = H\ XOR\ H_S$

Σ-Hash is distributed along with the stego-object M_S.

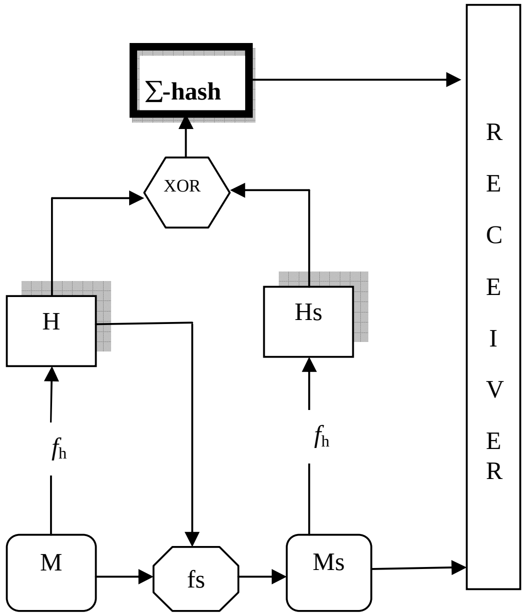


Fig. 3 : ∑-hash

Fig. 3: Depicts this ∑-hash procedure.

Verifying :

In order to verify the validity of the given hash functions three steps should be followed. Firstly, the given object, M_S is hashed in order to produce H_S' which is then XOR-ed with Σ-Hash:

$f_h(M_S) = H_S'$

$H' = \Sigma\text{-}Hash\ XOR\ H_S'$

As we mentioned in the previous section *H'* is used as the key for embedding random data to *M*. Thus, in order to retrieve *H*, which is stored as a secret message using steganography, the inverse steganographic function f_s^{-1} is performed, using *H'* as the stego-key:

$$f_s^{-1}(M_S, H') = H$$

Evidently, *H'* should be equal with *H*, otherwise an attack has been attempted.

VI. IMPROVED STEGANOGRAPHIC ALGORITHM

'Steganography is the art and science of communicating in a way which hides the existence of the communication. In contrast to cryptography, where the "enemy" is allowed to detect, intercept and modify messages without being able to violate certain security premises guaranteed by a cryptosystem, the goal of Steganography is to hide messages inside other "harmless" messages in a way that does not allow any "enemy" to even detect that there is a second secret message present' [Markus Kuhn 1995].

The best known stegnographic method that works in the spatial domain is the LSB [15] (Least Significant Bit), which replaces the least Significant bits of pixels selected to hide the information.

This paper proposes a new method called SLSB (Selected LSB), that improves the performance of the method LSB hiding information in only one of the three colors at each pixel of the cover image.

Description of the algorithm SLSB :

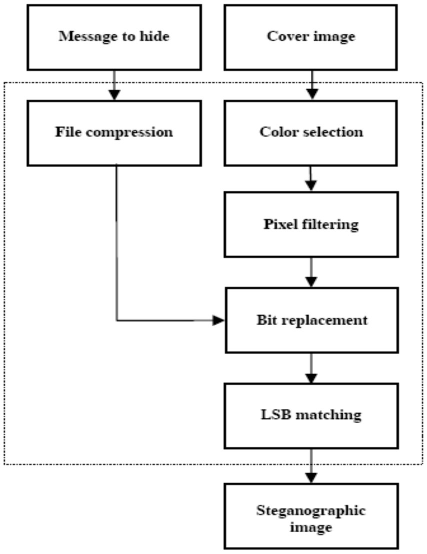


Fig. 4 : Structure of SLSB

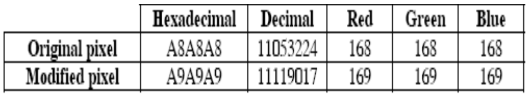
Most of the algorithms that work in the spatial domain using a LSB method (or any of its derivatives) as the algorithm for information hiding, that is, hide one bit of information in the least significant bit of each color of a pixel.

But these methods can't stand a type of statistical analysis (such as RS [16] or Sample Pairs [17]), even if partly camouflaged in the amount of information hidden.

The problem stems from the fact that modifying the three colors of a pixel produces a major distortion in the resulting color. This distortion not visible to the human eye but detectable by statistical analysis.

For example, if a pixel of the cover image with the RGB (Red-Green-Blue code) color A8A8A8 #is used, binary 10101000-10101000-10101000, and 1 bit with value 1 is set on each LSB bit of each color component, to hide the message 111, the result would be 10101001-10101001-10101001:

Table 1: Shows the results obtained by hiding the message 111 in the pixel 10101000 – 10101000 – 10101000 with the LSB method.

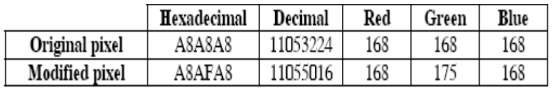


	Hexadecimal	Decimal	Red	Green	Blue
Original pixel	A8A8A8	11053224	168	168	168
Modified pixel	A9A9A9	11119017	169	169	169

In theory the three least significant bits of the pixel have changed, introducing a small distortion, but the difference between the old and new color represents a leap of 65793 colors in the scale of colors

One method that would introduce more efficiency and less distortion would store the 3 bits of information to hide in the same color. Using the same example, the 3 bits of information will be introduced in the 3 LSB bits of green color (10101000-10101**111**-10101000):

Table 2 : Shows results obtained hiding the message 111 in the pixel 10101000- 10101000- 10101000 with SLSB method



	Hexadecimal	Decimal	Red	Green	Blue
Original pixel	A8A8A8	11053224	168	168	168
Modified pixel	A8AFA8	11055016	168	175	168

In this case the leap in the scale of colors is 1792 colors (in the case of changing the color green, if modify the blue color difference would be only 7 colors), that being the extreme case because it has been replaced last 3 bits with 0 value for 3 bits with a 1 value, that is, in most cases the distortion will be much lower

Results :

To be able to compare the performance of this improvement on the LSB method, the image on Fig. 5

will be used as cover with BMP (Bit Mapped Picture) format and 512 x 512 pixels in size (24 bits /pixel).

Fig. 5 : Cover Image

To conclude the analysis of the new proposed algorithm its performance is compared with that from the best known and more used today steganography tools.

This comparison focuses on two aspects: the results of the RS and Sample Pairs analysis of steganographic images and the analysis of the results of the metrics of distortion

Table 3 shows a comparison of the results for the steganographic images obtained with the various tools in front of the RS analysis and the Sample Pairs analysis. The results of these analyses are an estimate of the percentage of hidden information. A lower ratio means a higher quality of the hiding method. percentage of hidden information. A lower ratio means a higher quality of the hiding method.

Table 3. Results obtained using a cover image of 786.486 bytes (Fig. 3) and a hidden message of 31.071 bytes (TXT file)

Tools	RS analysis	Sample Pairs analysis
Wbstego	14,33760	13,42652
JPHS	2,62679	2,68511
s-tools	2,30629	2,10435
Data privacy Tools	1,43103	0,96443
Original Image	0,67766	0,51907
SLSB	0,64431	0,47867

The results show that the new algorithm, offers among the best ever results (even reach a ratio below the original image, thereby preventing distinction between the original and steganographic image).

VII.ATTACKS TO ∑ -HASH

We consider an attacker that wishes to attack Σ-Hash in terms of collision resistance. Such an attacker would initially have two choices: find a collision for the hash of *M* or for the hash of M_S. In any case this would mean that *M’* or M_S’ should be found so that:

$$H' = f_h(M') = f_h(M) = H \quad \text{or}$$

$$H_S' = f_h(M_S') = f_h(M_S) = H_S$$

Considering the first choice, an attacker computes *M’* that produces the same hash value with *M*. In this case the embedding of *H* to *M’* would produce a significantly different stego-object $M_S' \neq M_S$. The SLSB steganographic algorithm ensure that the hidden data are not embedded into a specific area of the cover object but instead are equally and randomly spread into it. Thus, even slight variations in the contents of the cover object can produce different stego-objects. The hash value of a different stego-object would be different from Hs.

Similarly, an attacker may choose to find a collision for H_S by carefully choosing a different stego-object M_S’ so that: $f_h(M_S') = H_S$. In that case the inverse steganographic operation on M_S’ will give off a different secret message.

VIII. CONCLUSIONS

This paper focuses on the implementation of strong authentication methods using mobile phones. This paper introduced a new concept Σ-Hash which combines an efficient Steganographic algorithm called SLSB and hash functions in order to improve the authentication. We are working to implement Σ-Hash for real world applications. We have demonstrated that Σ-Hash can be used to avoid attacks to hash functions.

REFERENCES

[1] R. Rivest, The MD5 Message-Digest Algorithm. RFC 1321, IETF, April 1992.

[2] X. Y. Wang. The Collision attack on SHA-0. In Chinese, to appear on www.infosec.edu.cn, 1997.

[3] E. Biham, R. Chen, A. Joux, P. Carribault, W. Jalby and C. Lemuet. Collisions in SHA-0 and Reduced SHA-1. Advances in Cryptology–Eurocrypt’05, pp.36-57, May 2005.

[4] S. Katzenbeisser, F. A. P. Petitcolas, Information Hiding Techniques for Steganography and Digital Watermarking. Artech House, 2000.

[5] B. den Boer and A. Bosselaers, Collisions for the Compression Function of MD5. EUROCRYPT 1993, pp293–304.

[6] X. Wang, Y. Yin, H. Yu, Finding Collisions in the Full SHA-1. In Advances in Cryptology - CRYPTO '05, 2005.

[7] A. Lenstra, X. Wang and B. de Weger, Colliding X.509 Certificates, Cryptology ePrint Archive, Report 2005/067, 2005. Available at: http://eprint.iacr.org/

[8] M. Bellare, T. Kohno, Hash Function Balance and its Impact on Birthday Attacks. Advances in Cryptology-EUROCRYPT 04. Springer-Verlag, C. Cachin and J. Camenisch eds., 2004.

[9] Emmanouel Kellinis and Konstantinos Papapanagiotou, Using Steganography to Improve Hash Functions' Collision Resistance.

[10] Fadi Aloul, Syed Zahidi, Wassim El-Hajj, Two Factor Authentication Using Mobile Phones.

[11] Do van Thanh - Telenor & NTNU Strong authentication with mobile phone as security token.

[12] NIST, Secure hash standard. Federal Information Processing Standard, FIPS-180-1, April 1995.

[13] A. Lenstra, X. Wang and B. de Weger, Colliding X.509 Certificates, Cryptology ePrint Archive, Report 2005/067, 2005. Available at: http://eprint.iacr.org/

[14] M. Bellare, T. Kohno, Hash Function Balance and its Impact on Birthday Attacks. Advances in Cryptology-EUROCRYPT 04. Springer-Verlag, C. Cachin and J. Camenisch eds., 2004.

[15] Kurak, C. and McHugh, J.: A Cautionary Note on Image Downgrading. Proc. IEEE 8th Annual Computer Security Applications Conference. San Antonio, USA, Nov./Dec. 1992, pp. 153-155.

[16] Fridrich, J., Goljan, M. and Du, R.: Reliable detection of LSB steganography in color and grayscale images. Proc. ACM Workshop on Multimedia and Security, Ottawa, ON, Canada, Oct. 5, 2001, pp. 27-30.

[17] Dumitrescu, S., Wu, X. and Wang, Z.: Detection of LSB steganography via sample pairs analysis. 5th International Workshop on Information Hiding. Noordwijkerhout, Pays-Bas, 7/10/2002. Springer LNCS, vol. 2578, pp. 355-372, 2003.

❑❑❑