IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified

Vol. 5, Issue 9, September 2016

SVAP: Secure Visual Authentication Protocol

Sonal Sonyabapu Shinde[1], Ujwala H. Wanaskar[2]

Student, Dept of Computer Engineering, Padmabhooshan Vasantdada Patil College of Engineering, Pune, India[1]

Asst Professor, Dept of Computer Engineering, Padmabhooshan Vasantdada Patil College of Engineering, Pune, India[2]

Abstract: Keylogging is one of the harmful malware wherein the activity of recording the keys struck on a keyboard is being observed in such a way that the person using the keyboard is unknown about the fact that their actions are being observed. This paper aims to prevent keylogging attacks by assigning proper authentication codes. The methodology of this research has progressed using System model, Linear and Matrix Barcodes, Message signing and Visual Signature Validation. Demonstration of careful visualization design enhancing the security and the usability of authentication is being successfully reflected in this paper. This research enables the user to store essential information in an encrypted format which can be decrypted very speedily thereby enabling to achieve a high level of usability while satisfying stringent security requirements using strict authentication.

Keywords: Keylogging; Authentication; Encryption; Decryption.

I. INTRODUCTION

The loss and steal of devices is getting a big problem because the data are not secured properly.[1] Keylogging or keystroke logging is a harmful malware in which an activity of recording the keys struck on a keyboard, normally in a secretive way, is performed so that the person using the keyboard is unknown about the fact that their actions are being observed.[2] Keylogging attacks or those that utilize session hijacking, phishing and pharming and visual fraudulence, cannot be addressed by simply enabling encryption.[3] Keyloggers malignantly track customer information from the comfort attempting to recuperate individual and private information.[4] Nowadays, there are many threats against electronic and financial services which can be classified into two major classes: credential stealing and channel breaking attacks. Credential stealing is nothing but username, password and pin number which can be stolen by the attacker if they are poorly managed. Channel breaking attacks is nothing but eavesdropping on communication between users and a financial institution.[5, 6, 7]

There are two types of keyloggers, hardware keylogger and software keylogger. Hardware keylogger used for keystroke logging is a method of recording victim's keystrokes which will include ATM PIN, login password etc. They can be implemented by BIOS-level firmware or may be used through a device plugged in line between a computer keyboards and a computer. Software keyloggers logs and monitors the keystrokes and data within the target operating system, store them on hard disk or in remote locations, and send them to the attacker. Software keylogger monitoring is mainly based on the operating-system.[8]

A keylogger is a software designed to capture all of a user's keyboard strokes and then make use of them to impersonate a user in financial transactions. The threat of such keyloggers is pervasive and can be present both in personal computers and public kiosks. The weakest link in software-based full disk encryption is the authentication procedure today.[9] The worst part is that, keyloggers, often root kitted, are hard to detect since they will not show up in the task manager process list. To mitigate the keylogger attack, virtual or onscreen keyboards with random keyboard arrangements are widely used in practice. Both techniques, by rearranging alphabets randomly on the buttons, can frustrate simple keyloggers. Unfortunately, the keylogger, which has control over the entire PC, can easily capture every event and read the video buffer to create a mapping between the clicks and the new alphabet. Another mitigation technique is to use the keyboard hooking prevention technique by perturbing the keyboard interrupt vector table. However, this technique is not universal and can interfere with the operating system and native drivers. It is not enough to depend only on cryptographic techniques to prevent attacks which aim to deceive user's visual experience while residing in a PC. Human user's involvement in the security protocol is sometimes necessary to prevent this type of attacks but humans are not good at complicated calculations and do not have a sufficient memory to remember cryptographically strong keys and signatures.[5] The protection against keylogger addresses the problem of programs being able to read the global key state or the actual key buffer of a window. It does so by installing a filter driver in the kernel which receives every keystroke before it is sent to the Windows driver. This enables keystrokes to be filtered out as if they had never occurred. The result is that the keystroke appears in neither the global key state nor the key buffer, thus preventing malware from intercepting the input data. However, so that the keystrokes are not simply filtered out, the keys that have been pressed are obviously then added back into the system by sending them directly to the foreground

DOI 10.17148/IJARCCE.2016.5974

window. This side channel ensures that Windows cannot determine that a particular key has been pressed. Windows simply knows that input has occurred in the foreground window. [10]

The concept behind keylogger protection is shown in fig 1;

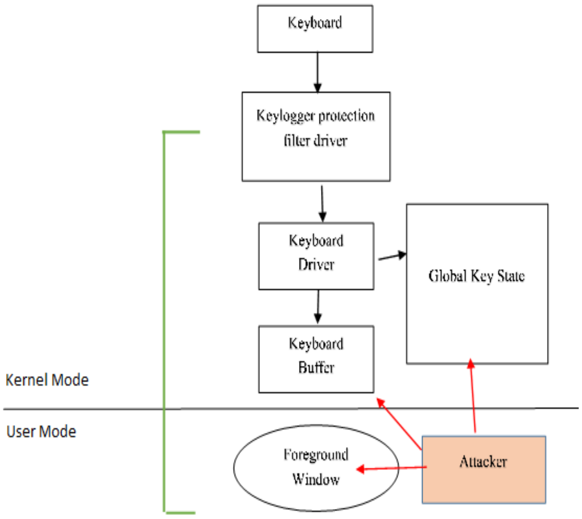

Fig 1: Processing keyboard input in Windows and the concept behind Keylogger Protection [10]

II. MOTIVATION

Keylogging exhibits an extraordinary test to security supervisors. Dissimilar to customary worms and viruses, certain sorts of keyloggers are everything except difficult to discover. Keyloggers are a kind of malware that malignantly track customer information from the comfort attempting to recuperate individual and private information. Growing machine use for essential business and individual activities using the Internet has made feasible treatment of Keylogging basic.

III. PROPOSED SYSTEM AND DESIGN

A] Problem Definition:

1) Two protocol for authentication that uses visualization by technique for increased reality to give both high security and high convenience. We exhibit that these conventions are secure under a couple of certifiable attacks including keyloggers. Both conventions offer great circumstances in light of visualization both as far as security and convenience.

2) Model utilization as Android applications which demonstrate the convenience of our conventions in true organization settings.

B] System Architecture

Our approach to solving the problem is to introduce an intermediate device that bridges the human user and a terminal. Then, instead of the user directly invoking the regular authentication protocol, she invokes a more sophisticated but user-friendly protocol via the intermediate helping device. Every interaction between the user and an intermediate helping device is visualized using a Quick Response (QR) code. The goal is to keep user-experience the same as in legacy authentication methods as much as possible, while preventing keylogging attacks. Below Fig.2 shows the architecture of the system. It gives the idea of working of the system.

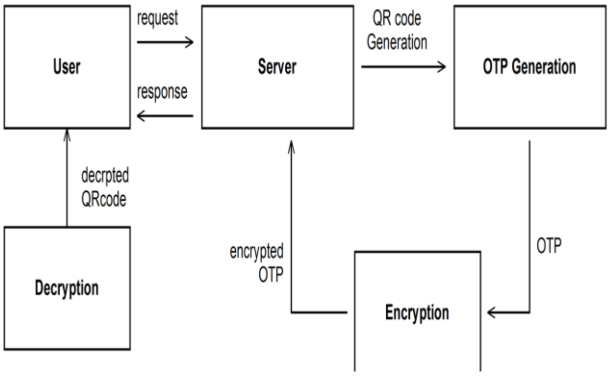

Fig 2: System Architecture

C] Mathematical Model and Design

Let W be the whole system which consists

Input = {U, M, C, k, S, Pvk, Pbk, M}.

a) Let u is the set of number of users.

U = {u1, u2, …, un}.

b) k is the secret key used for encryption.

c) M is the message sent from the set M.

d) C is the cipher-text in the set C.

e) S is the signature generated for sending message.

f) Pvk is the private key.

g) Pbk is the public key.

Functions:

1. Encrk (.): an encryption algorithm which takes a key k and a message M from set M and outputs a cipher-text C in the set C.

2. Decrk (·): a decryption algorithm which takes a ciphertext C in C and a key k, and outputs a plain-text (or message) M in the set M.

3. Sign (·): a signature generation algorithm which takes a private key Pvk and a message M from the set M, and outputs a signature σ.

4. Verf (·): a signature verification algorithm which takes a public key Pbk and a signed message (M, σ), and returns valid or invalid.

5. QREnc (·): a QR encoding algorithm which takes a string S in S and outputs a QR code.

6. QRDec (·): a QR decoding algorithm which takes a QR code and returns a string S in S.

IV. METHODOLOGY

Modules Information

a) System Model

b) Linear and Matrix Barcodes

c) Message signing

d) Prevention of Session Hijacking with Visual Signature Validation

Module(s) Description

a) System Model

It consists of four different entities (or participants), which are a user, a Smartphone, a user's terminal, and a server. The user is an ordinary human, limited by human's shortcomings, including limited capabilities of performing complex computations or remembering sophisticated cryptographic credentials, such as cryptographically strong keys. With a user's terminal such as a desktop computer or a laptop, the user can log in a server of a financial institution (bank) for financial transactions. Also, the user has a Smartphone, the third system entity, which is equipped with a camera and stores a public key certificate of the server for digital signature verification. Finally, the server is the last system entity, which belongs to the financial institution and performs back-end operations by interacting with the user (terminal or Smartphone) on behalf of the bank.

b) Linear and Matrix Barcodes

A barcode is an optical machine-readable representation of data, and it is widely used in our daily life since it is attached to all types of products for identification. In a nutshell, barcodes are mainly two types: linear barcodes and matrix (or two dimensional, also known as 2D) barcodes. While linear barcodes have a limited capacity, which depends on the coding technique used that can range from 10 to 22 characters, 2D barcodes have higher capacity, which can be more than 7000 characters. For example, the QR code a widely used 2D barcode can hold 7,089 numeric, 4,296 alphanumeric, or 2,953 binary characters, making it a very good high-capacity candidate for storing plain and encrypted contents alike.

c) Message signing

For the generality of the purpose of this protocol and the protocols, and to prevent the terminal from misrepresenting the contents generated by the server, one can establish the authenticity of the server and the contents generated by it by adding the following verification process. When the server sends the random permutation to the user, it signs the permutation using the server's private key and the resulting signature is encoded in a QR code. Before decrypting the contents, the user establishes the authenticity of the contents verifying the signature against the server's public key. Both steps are performed using the Sign and Verf algorithms. Verification is performed by the smart phone to avoid any man-in-the-middle attack by the terminal.

d) Prevention of Session Hijacking with Visual Signature Validation

1) A user requests via terminal to the server money transfer denoted as T that describes sender name/account, recipient name/account, a timestamp, and amount of money to transfer.

2) The server checks the ID to retrieve the user's public key (PKID) from the database. Then, it picks a fresh OTP to prepare QR = QREnc(EOTP ; T; _ = Sign(PrK; T)), where PrK is a signing key of the server. Then, it sends QR to the user to authorize the transaction.

3) On the terminal, a QR code QR is displayed prompting the user to type in the OTP string.

4) The user decodes the QR code to get (EOTP = QRDec(QREOTP); T; _) with her smartphone application. Here the application verifies the time stamp and the signature by Verf(PubK; T; _) to show the result (Valid/Invalid) on the screen with the decrypted OTP and T. If the application fails to validate the signature, it doesshow neither the decrypted OTP nor T, but displays an error message to alert the user. When the user is confirmed with the signature verification result and with T, she inputs the OTP to the terminal, which is sent back to the server.

5) The server checks the result and if it matches with the OTP that the server has sent earlier, the user is authenticated. Otherwise, the user is denied.

V. RESULT AND ANALYSIS

The results of this research obtained by using the above methodology is represented as follows;

Analysis 1

Table 1: QR Code Generation Time

QR Codes	QR Code Generation Time (sec)
QR1	47
QR2	47
QR3	56
QR4	34
QR5	53
QR6	32
QR6	64

Table 1 gives the information about time taken for generation of QR code by the system at the time of registration by user. Graphically it is shown in fig 3. The QR code get generated in very less amount of time which saves the information very securely. The user requires decryption of these QR code to know the information stored in it.

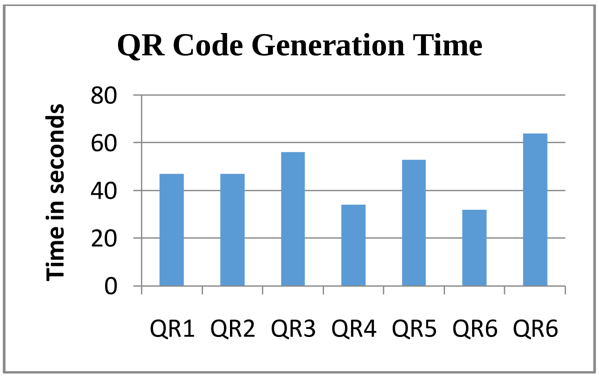

Fig 3: QR Code Generation Time

DOI 10.17148/IJARCCE.2016.5974

Analysis 2:

Table 2: QR Code Decryption Time

QR Code	QR Code Decryption Time (sec)
QR1	6
QR2	7
QR3	10
QR4	8
QR5	8
QR6	5
QR6	12

Table 2 gives the information about time taken for decryption of QR code by the system at the time of decryption by user. Graphically it is shown in fig 4. The system takes few seconds of time to decrypt the QR code and user can retrieve the required information stored in it. The only authenticated user has authority to decrypt these QR code so that the confidentiality of the information can be maintained properly.

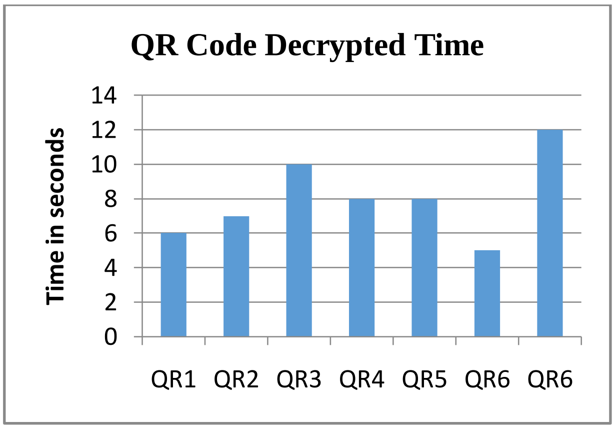

Fig 4: QR Code Decryption Time

Analysis 3:

Table 3: Comparison of QR code Generation Time and QR Code Decryption Time

	QR Code Generation Time (sec)	QR Code Decryption Time (sec)
QR1	47	6
QR2	47	7
QR3	56	10
QR4	34	8
QR5	53	8
QR6	32	5
QR6	64	12

Table 3 gives the information about QR code Generation Time and QR Code Decryption Time. From above data it can be interpreted that the time required for generation of QR code is less than the time required for decryption of QR code. Within few seconds user can retrieve the information from decrypted QR code which will help in fast processing further. Graphically it is shown in fig 5.

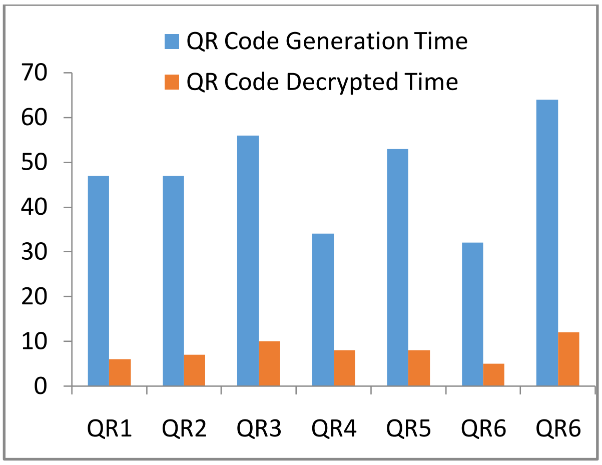

Fig 5: Comparison of QR Code Generation Time and QR Code Decrypted Time

VI. COMPARITIVE ANALYSIS

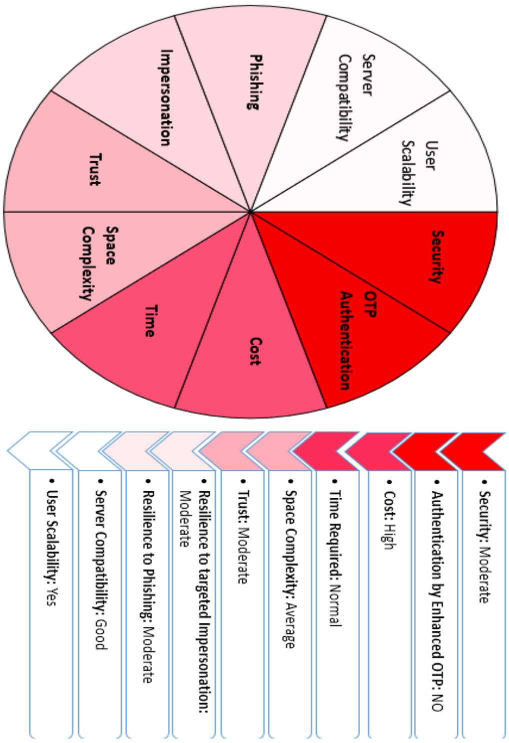

Fig 6: Existing Security System

Scale: Each segment in the graph represents 10 percent weightage
Color: Color intensity defines the importance of parameters, important being the darkest. Red color shows the negative aspects of existing system as compared to Green color which shows positive aspects of the proposed system.

DOI 10.17148/IJARCCE.2016.5974

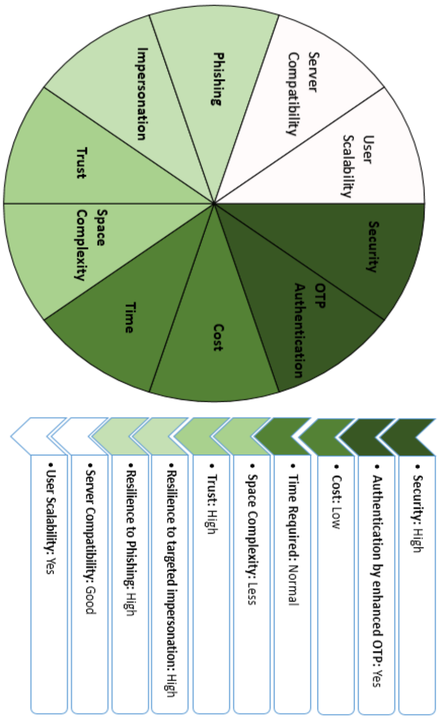

Fig 7: Proposed Security System

VII. CONCLUSION & FUTURE SCOPE

This research article attempts to an insight on the recent advancements on the attempts to mitigate the risks of keylogging attacks. It is a tool for the speedy encryption and decryption of data required in emergency situation thereby maintaining its confidentiality also. It may pave a path to help future advancements in the areas related to keylogging attacks. The author also propose that much there is still scope to perform inventory work in the area of keylogging attacks which needs to be addressed and worked upon in the coming years.

ACKNOWLEDGMENT

The author wishes to acknowledge my friends and colleagues who have extended their support during this research work. I would also like to thank my family members for their tiring efforts during my research.

REFERENCES

[1] M. Trojahn and F. Ortmeier, "Biometric authentication through a virtual keyboard for smartphones", International Journal of Computer Science & Information Technology, Vol. 4, No. 5, Page 1-12, Oct 2012

[2] D. Bhave, P. Bhavsar, S. Chavan and K. Gore, "Keylogging-resistant visual authentication protocol", International Journal of Advanced Research in Computer and communication Engineering, Vol 5. Issue 2, Page 520-524, Feb 2016

[3] S. P. Goring, J. R. Rabaiotti and A. J. Jones, "Anti-keylogging measures for secure internet login: an example of the law of unintended consequences", Computers & Security, Page 1-9, Feb 2007

[4] P. K. Veni and B. Naresh, "A novel visual authentication protocols implementation based on keylogging-resistant", International Journal of Scientific Engineering and Technology Research, Vol. 4, Issue 28, Page 5470-5477, Jul 2015

[5] D. Nyang, A. Mohaisen and J. Kang, "Keylogging-resistant visual authentication protocols", IEEE Transactions on Mobile Computing, Vol. 13, No. 11, Page 2566-2579, Nov 2014

[6] R. Sangeetha, N. H. Vinodha and A. V. Kalpana, "QR code based encrypted matrix representation for eradication hardware and software keylogging", International Journal of Engineering Sciences and Research Technology, Page 642-647, Apr 2015

[7] R. Saraswathi, G. Shanmathi, P. Preethi and U. Arul, "Secure internet banking with visual authentication protocol", International Journal of Scientific Research in Science, Engineering and Technology, Vol. 1, Issue 1, Page 351-353, Jan-Feb 2015

[8] H. Pathak, A. Pawar and B. Patil, "A survey on keylogger-A malicious attack", International Journal of Advanced Research in Computer Engineering and Technology, Vol. 4, Issue 4, Page 1465-1469, Apr 2015

[9] T. Muller, H. Spath, R. Mackl and F. C. Freiling, "STARK-Tamperproof authentication to resist keylogging", Chapter: Financial Cryptography and Data Security, Volume 7859 of the series lecture notes in Computer Science, Page 295-312

[10] Keylogger protection-System security research, GData, Whitepaper, Page 1-8, Mar 2014

BIOGRAPHIES

Ms. Sonal Sonyabapu Shinde, Student, Department of Computer Engineering, Padmabhooshan Vasantdada Patil Institute of Technology, Pune.

Ms. Ujwala Wanaskar, Assistant Professor, Department of Computer Engineering, Padmabhooshan Vasantdada Patil Institute of Technology, Pune.

DOI 10.17148/IJARCCE.2016.5974