iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 3 **Issue:** IV **Month of publication:** April 2015

DOI:

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)

Secure Internet Banking With Visual Authentication Protocols

Vishrut Sharma[1], C Jawahar Sandeep[2], Nikhil Katariya[3], Aruna. P[4]
Dept. of Computer Science & Engineering, SRM University, Chennai, Tamilnadu, India.

Abstract*—Keylogging is an activity of capturing users' keyboard strokes and records the activity of a computer user in a covert manner using keylogger hardware and software. The keyloggers secretly monitors and log all keystrokes. Unlike other malicious programs, keyloggers do not cause any threat to system. But it can be used to intercept passwords and other confidential information entered via the keyboard by considering various rootkits residing in PCs (Personnel Computers) that breaches the security. Cyber criminals can get user names, email passwords, PIN codes, account numbers, email addresses, passwords to online gaming accounts, e-payment systems, etc. As a result, it impersonates a user during authentication in financial transactions. To prevent keylogging, the strict authentication is required. The QR code can be used to design the visual authentication protocols to achieve high usability and security. The two authentication protocols are Time based One-Time-Password protocol and Password-based authentication protocol. Through accurate analysis, the protocols are proved to be robust to several authentication attacks. And also by deploying these two protocols in real-world applications especially in online transactions, the strict security requirements can be satisfied.

***Keywords*— keylogging; phishing; pharming; session hijacking; QR code; authentication; malicious code; attack; android; visualization**

I. INTRODUCTION

The credential stealing and channel breaking attacks [9] are the two major threats in electronic and financial services. The credential information such as users' identifiers, passwords, and keys can be easily stolen by the attacker from the target computers if they are less secured. On the other hand, channel breaking attacks allow eavesdropping of communication between the user and the financial institution [10]. But the classical channel breaking attacks can be prevented by using IPsec [4] and SSL security channels. The recent channel breaking attacks are more challenging like keylogging which utilizes session hijacking, pharming, phishing and visual fraudulence. It is difficult to prevent these attacks by simply encryption. For example, if a personal computer is infected with malicious software, then it is an easy target for credential attackers. Keyloggers are used as an surveillance tool by the employers to ensure employees use work computers for business purposes only. Unfortunately, keyloggers are embedded in spyware and allows the information to be sent to unkown third party. Keyloggers can be used in some IT organizations to troubleshoot technical problems in computers and business networks. Keyloggers are also used by a family or business to monitor the people without their knowledge. Finally, keyloggers are installed in public kiosks to steal credit card information or passwords. Keylogging allows malicious software to capture keyboard strokes whenever the user types in the specific application or forms to obtain the passwords. Cyber criminals use keylogging to capture credentials and authentic information to hack the account and performs financial fraudulance and therefore gains access to confidential information. Malware uses several techniques to log keystrokes such as hooking into the keyboard driver and other operating system services. The keylogger is present both in personal and public computers and it is pervasive. Keyloggers are often root kitted. So the presence of keyloggers cannot be detected. To overcome keylogging attack, virtual or onscreen keyboards are used. Both the techniques rearrange the alphabets randomly and therefore frustrate simple keyloggers. But the keylogger has control over the entire PC, which can capture every event and read the video buffer to create a mapping between the clicks and new alphabet. The keylogging attack is quite similar to the shoulder- surfing attack where the attacker sees the direct input of the client to the computer and also every behavior of the client. Many graphical password schemes are introduced in [6], [8], [12] to prevent shoulder-surfing attack. But many of these schemes are unusable. Even though the cryptographic secrets are securely delivered to the client's PC, the attacker residing on the client's PC can easily observe and deceive the information. To solve this problem, the intermediate device between human and terminal is introduced. This helps to design a human involving protocol. Every interaction between the client and the intermediate device is visualized using Quick Response (QR) code. In these protocols, the client does not need to memorize any information other than password and PIN. However, the authentication process can be visualized which enhances security and usability to the client. The security protocol has the client involvement using smartphone with augmented reality. A smartphone with camera is used to visualize the authentication process.

©IJRASET 2015: All Rights are Reserved

Instead of implementing the entire security protocol in computer, a part of it is moved to the smartphone. This visualization in smartphone offers protection against malware, keylogging attacks and shoulder-surfing attacks.

A. Scope and Contributions

The two authentication protocols are time-based one-time- password protocol and password-based authentication protocol. Through accurate analysis, one can prove that these two protocols are resistant against many challenging attacks that are applicable in other protocols specified in the literature. The two protocols are secure under many real-world attacks that visualize the authentication process to enhance both security and usability. The prototype implementation in the form of Android smartphone applications demonstrates the usability of protocols in real world deployment. The visual authentication protocols can be used in ATM (Automatic Teller Machine) and public computers which involves financial transformations. Moreover, it does not need any channel between the server and the smartphone.

B. Types of Keyloggers

Keyloggers are a serious security threat that can be extremely harmful to both businesses and consumers.The keylogging attack can be performed either using software keyloggers or hardware keyloggers.

1) *Software-based Keyloggers:* The keylogging software is a type of surveillance software which is installed into the target computers and the presence of software keyloggers cannot be felt since it is not shown in the task manager. The keylogger creates the log file for every session which is sent to the specified receiver. This danger was recently highlighted when Sumitomo Mitsui Banking Corporation discovered a keylogger installed on its network in London [1]. There have been other high-profile cases in keylogging attack. In 2003, the perpetrator installed the software at more than 14 Kinko locations in New York and using it to open bank accounts with the names of some of the 450 users whose personal information he collected [2]. Also in 2003, Valve Software founder Gabe Newell found the source code to his company's Half-Life 2 game stolen after someone planted a keylogger on his computer [3]. Some of the software-based keyloggers are hypervisor- based, API-based, kernel-based, form grabbing based, memory injection based, packet analyzers, and remote access software keyloggers.

2) *Hardware-based Keyloggers:* The hardware-based keyloggers are the hardware devices like USB or pendrive which does not need any support of software components to work on computer. The hardware keylogger may be injected into the public computers without the knowledge of the user to monitor the behavior of the users. The KeyGrabber USB is an USB hardware keylogger with a 16 MB or 2 GB internal flash disk which is organized as a file system. Any information typed on the keyboard will be captured by the KeyGrabber USB and stored on the internal Flash Drive in a special file. This captured data may be retrieved on any other computer containing a USB port and keyboard, by switching into Flash Drive mode. The keylogger gives instant access to all captured data and pop up as a removable drive. The KeyGrabber USB is 100% transparent for computer operation which does not requires software or drivers. Some of the hardware-based keyloggers are firm-based, keyboard hardware, wireless keyboard sniffers, keyboard overlays, acoustic keyloggers, electromagnetic emissions, optical surveillance, and physical evidence.

C. Organization

The rest of this paper is organized as follows. In section II, the system, trust and attacker models, and comparison of linear barcode and QR code are explained. In section III, the working of two visual authentication protocols is briefly explained. In section IV, several issues related to two protocols are explained. Section V reviews the related works from the literature. In section VI, the conclusion is made.

II. SYSTEM AND THREAT MODEL

A. System Model

The system model comprises of four different entities such as a client, a smartphone, a client's terminal (PC) and a server. The client is a user or an ordinary human with limited capabilities of remembering cryptographic credentials such as keys and performing complex mathematical computations. A client's terminal is a client's PC which is used to connect to a server for performing financial transactions. The client has the smartphone which stores the public key certificate of the server or digital certificate equipped with a camera. The server is the system entity belongs to the financial institution which interacts with the user by performing all the back-end operations.

©IJRASET 2015: All Rights are Reserved

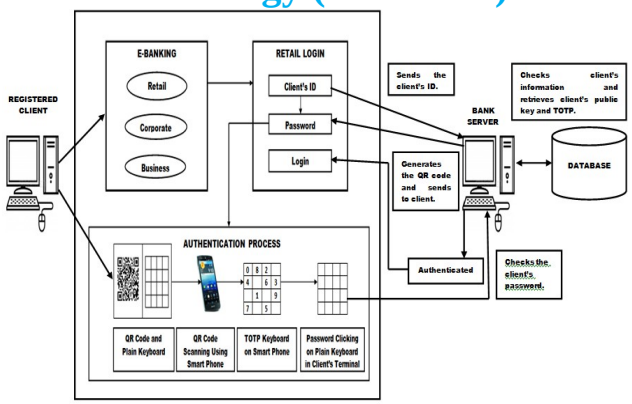


Fig. 1. shows the overall system architecture. Here, the e-banking is taken as an example to show how the authentication process works.

The client or an user is registered in a particular bank for performing online transactions and provided with the unique client ID and password. The registered client can log on to particular bank site. The client must enter into retail login. When the client sends the unique ID to the server, the server checks the client's information from the bank database. If the client's information is correct, the server retrieves the public key and fresh random time-based one-time-password (TOTP) from the database.The server generates the QR code which comprises of unique client ID, public key, TOTP and time slot. Then the QR code is sent to the client. On client's terminal, the QR code is displayed. Now, the client has to take his smartphone in which the QR code scanning application is already installed. The QR code has to be scanned. After scanning the QR code, the decoded information will be displayed in the smartphone. The randomized keyboard which looks like a 4x4 matrix with random arrangements of 0-9 digits is displayed in the smartphone. On the client's terminal the password box is replaced with the 4x4 blank keyboard matrix. Now, the client has to just click on the rows or columns of the blank keyboard matrix by seeing where is password has been arranged in the smartphone. From the client's terminal, only the ID of the keyboard matrix is sent to the server. The server also does not know the password of the client. Based on the ID of the keyboard matrix, the client gets authenticated. If the client clicks on the wrong ID, again the previous steps are repeated by sending a newly generated QR code to the client. And also if the client fails to login within the allotted time slot, the server will automatically generates a new QR code with new TOTP. After the client gets authenticated, the client can enjoy all the e-banking services.

B. Trust and Attacker Models

The following must be assumed to ensure that the entities of the system are secure and trusted. First, the channel between server and client's terminal is secured with an SSL or HTTPS connection. Second, the server is assumed to be immune to several attacks. So, the attacker concentrates on the client. Third, the keylogger attacker resides on the client's terminal. The attacker is capable of capturing the security of the system. The attacker has full control over the client's terminal. So,

1) The attacker can capture client's credential information such as password, private key and TOTP.

2) The attacker can perform session hijacking by showing a fake genuine looking webpage in financial transactions. Therefore the authenticated session can be hijacked by the attacker.

C. Comparison of Quick Response Code with Linear Barcode

QR code is developed by Japanese Denso Wave corporation in 1994. It is a two dimensional barcode. There are 40 versions and four levels of error correction in QR code. The barcodes are attached to all sort of products for identification which is a optical machine-readable representation of data. Linear barcodes are one dimensional and have a limited capacity of coding 10 to 22 characters. The QR code has the high capacity which can hold 7,089 numeric, 4,296 alphanumeric, and 2,953 binary characters [1]. QR Code has been approved as an AIM Standard, a JIS Standard and an ISO standard. So QR Code is being used in a wide variety of applications, such as manufacturing, automotive, logistics, sales, and other business applications. The QR code has the efficiency to decode all types of information such as website URL, contact address, phone number, geographical location, a text message, s calendar event, etc. some of the features of QR code are given as follows:

*1)*High capacity encoding of data
*2)*High-speed reading
*3)*Chinese encoding capability

©IJRASET 2015: All Rights are Reserved

4) Readable from any direction from 360 degree
5) Dirt and Damage Resistant
6) Structured Append Feature



Fig. 2. Barcode



Fig. 3. QR code

At first, the QR code has been designed to be used in automotive industries. But now, it has been widely used in the advertisement so that a client can use the smartphone and scan to know more information about the advertised products. The barcode scanner applications are created which is compatible for smartphones like android and ios.

III. QR-BASED VISUAL AUTHENTICATION

In this section, two visual authentication protocols are explained. Before getting into the protocols, it is necessary to know about the algorithms used in the proposed system. The algorithms are explained as follows.
The public key encryption scheme with IND-CCA2 IN Distinguishability against Adaptive Chosen Ciphertext Attacker) security would be good for the proposed system application. IND-CCA2 makes the cipher text different even though the plaintext is the same whenever encrypted by adding random padding to a plaintext [11]. This restriction will prevent an attacker from checking whether his guess for the random layout is right or not. Thus, the security of the scheme is not dependent on the number of possible layouts but the used encryption scheme. If no such encryption is used, the adversary will be able to figure out the layouts used because he will be able to verify a brute-force attack by matching all possible plaintexts to the corresponding cipher text. On the other hand, when such encryption is used, the 1-1 mapping of plaintext to cipher text does not hold anymore and launching the attack will not be possible at the first place. Also, any signature scheme with EUF-CMA (Existential-UnForgeability against adaptive Chosen-Message Attacker) can be used to serve the purpose of proposed system. For details on both notions of security, see [7]. In particular, and for efficiency reasons, the short signature is recommended [2].

A. Time-based One-time-password protocol

In this section, a Time-based One-time-password authentication protocol is introduced which is referred as first protocol. It make use of random string for authentication. This protocol works as follows:

1) The client sends the unique client ID to the server.
2) The server checks the client's information from the database and retrieves the client's public key (PK_{ID}).
3) The server then picks a fresh random string TOTP with a time slot and encrypts it with the public key to obtain

$$E_{TOTP} = Encr(PK_{ID}(TOTP)). \quad (1)$$

The server generates the QR code and sends it to the client.
4) In the client's terminal, a QR code QR_{EOTP} is displayed.
5) The client decodes the QR code with

$$E_{TOTP} = QRDec(QR(E_{TOTP})). \quad (2)$$

6) The random string is encrypted with client's public key (PK_{ID}), the client can read the TOTP string only through her smartphone by and type in the TOTP in the terminal with a physical keyboard.

©IJRASET 2015: All Rights are Reserved

$$TOTP = Decrk(E_{TOTP}) \quad (3)$$

The client has to type the TOTP in the terminal where the keyboard matrix is displayed.

7) The server checks the result entered by the client and if it matches what the server has sent earlier, the client is authenticated.
8) If the client does not authenticated, the access is denied.

B. Password-based authentication protocol with randomized onscreen keyboard

In this section, the second protocol password-based authentication protocol is described. Here, the password is shared between server and client, and a randomized keyboard. The protocol works as follows:

1) The client connects to the server and sends unique client ID to the server.
2) The server checks the received unique client ID to retrieve the client's public key (PK_{ID}) from the database.
3) The server prepares a random permutation of a keyboard arrangement, and encrypts it with the public key to obtain

$$E_{KBD} = Encr(PK_{ID}(\pi)). \quad (1)$$

4) The server encodes the ciphertext with QR encoder to obtain

$$QR(E_{KBD}) = QR(Enc(E_{kID}(\pi))). \quad (2)$$

5) The server sends the result to the client with a blank keyboard.
6) In the client's terminal, a QR code ($QR(E_{KBD})$) is displayed together with a blank keyboard.
7) The onscreen keyboard does not have any alphabet on it, the client cannot input her password.
8) The client executes her smartphone application which first decodes the QR code by applying to get the ciphertext (E_{KBD}).

$$QR(Dec(QR(E_{KBD})) \quad (3)$$

9) The ciphertext is then decrypted by the smartphone application with the private key of the client to display the result on the smartphone's screen

$$\pi = Decr(SKID(EKBD)). \quad (4)$$

10) When the client sees the blank keyboard with the QR code through an application on the smartphone that has a private key, alphanumerics appear on the blank keyboard and the client can click the proper button for the password.
11) The client types in her password on the terminal's screen while seeing the keyboard layout through the smartphone.
12) The terminal does not know what the password is but only knows which buttons are clicked.
13) Identities of the buttons clicked by the client are sent to the server by the terminal.
14) The server checks whether the password is correct or not by confirming if the correct buttons have been clicked.

IV. DISCUSSION

In this section, the several issues related to two protocols are discussed. Some of the issues are session hijacking, keylogging, transaction verification, securing transactions, visual channels and visual signature validation.

TABLE I
COMPARISON OF TWO PROTOCOLS AND THEIR RESISTANCE TO DIFFERENT ATTACKS WHEN THE TERMINAL AND THE SMARTPHONE ARE UNDER CONTROL OF THE ATTACKER

Protocol	Attack	Brute-force	Keylogger	Malware
TOTP	Smartphone	Yes	Yes	No
	Terminal	Yes	Yes	Yes
PAP	Smartphone	Yes	Yes	Yes
	Terminal	Yes	Yes	Yes

©IJRASET 2015: All Rights are Reserved



c) Clicking password on blank keyboard

Fig. 4. Shows the expected outcomes of the authentication protocols, (a) shows the QR code scanning at the terminal using smartphone. (b) shows the encoded keyboard layout on the smartphone with the random arrangements of digits. (c) shows the blank keyboard layout at the terminal which is prompting to click the password.

A. Preventing Session Hijacking

The attacker can hijack the authentication session through a malware immediately after the client inputs a password. The session hijacking can be detected by sending the transaction related information to the server using the smartphone triggered by the client through a side channel introduced in section E. Just after running the password input procedure, the smartphone application may send additional information on the client's transaction request that is signed by the smartphone's private signing key and/or encrypted with the server's public key through cellular network.

B. Preventing Keylogging

The visual authentication protocols aims at preventing keylogging attacks. The client just clicks the password on the blank keyboard. So, only the identities of the blank keyboard are sent to the server. The keylogger cannot record the keystrokes of the client because the mouse is used for clicking the password. Some keyloggers has the capacity of taking the snapshot of the client's screen to steal the credential information. But the keyloggers has nothing to do with this type of system since the only the blank keyboard is displayed in the client's terminal and the password is not visible. Even though the client's mouse clicks are captured using snapshots, the attacker cannot impersonate the password. Because the attacker cannot guess the password using the mouse clicks. The password is randomly arranged and hidden in the blank keyboard.

If the keylogger software is installed in the smartphone, it can also capture all the client's keystrokes on smartphone. But the client does not uses any key of smartphone for authentication. The smartphone displays only the scanned QR code i.e. randomly arranged 4x4 keyboard layout and if the snapshot is taken, there is no use of it. Because the keyboard layout varies every time when the client scans the QR code during authentication process.

C. Transaction Verification

Assume that a given client is visiting a banking server and is about to transfer money to other account. Even if the client's terminal is infected with some malicious ware or the client is visiting a phishing site, she cannot recognize it easily, because the HTML page that the client is watching is visually the same as the genuine page. Even when a bank server asks the client to input credentials such as a password, a one-time-password generated from a hardware token, and a certificate based signature to confirm the transaction, the client is willing to input her credentials as requested, and with the credential information the attacker is able to prepare a valid transfer request to her account. In this section, an effective and useable approach to defeat these visual fraudulence with the aid of a smartphone are shown. Similar in essence to the previous discussed protocols, the protocol for preventing the visual fraudulence consists of the following:

©IJRASET 2015: All Rights are Reserved

1) The client sends an HTML page to request money transfer, for instance. This page might include a receiver's account number, the amount of money to be transferred, and the receiver's name, etc.

2) The server responds with confirmation HTML page along with a QR code that includes transaction information and a digital signature of messages in the HTML page. The page might include a code to prompt a client to input credentials to confirm the order.

3) The client reviews the HTML page received from the server then with an application on her smartphone she takes a snapshot of the QR barcode.

4) The application on the smartphone decodes and verifies the digital signature over the attached message with the server's public key. If the signature is valid, it will show the message with a mark indicating the signature was verified. Otherwise, it will warn the client with a mark indicating invalid signature. The mark may be a background color (green on a valid signature and red on an invalid one) or simple warning words.

5) The client checks if the message matches with the one in the HTML page and the signature is verified. If it is valid and correctly verified, the client continues to confirm the transaction by inputting her credential.

D. Securing Transactions

Financial transactions are usually secured by encrypting all transaction-related information during the transmission. In many cases, the encrypted information should be decrypted at the terminal (client's PC, most likely, or a PC at public place) to be shown to the client. However, under the assumption that there is a malware inside the terminal, the attacker does not need to break the cipher, but is enough to read the information after being correctly decrypted. The encrypted channel is established just between the server and the client's terminal. To make transactions more secure, it is needed to extend the encrypted channel beyond the client's terminal. Accordingly, instead of decrypting the cipher text at the client's terminal, decrypt it at the smartphone.

E. Visual channels

1) Backward Visual channel from PC to smartphone: The use of the visual channel to input encrypted credentials from the smartphone to the terminal has an interesting security implication. As is the case with using e-banking on untrusted terminals, imagine that such terminal is infected with a virus, or has a malware, which could be a keylogger. If the client is to use the authentication credentials directly on the terminal, it is obvious that these credentials will be compromised. On the other hand, if these credentials are keyed in on the smartphone and to be transferred using the visual channel between the smartphone and the terminal in an encrypted form for the server, the keylogging attacker will be prevented from logging these credentials on the terminal.

2) Replacing Visual channels with Bluetooth: The visual channel in Protocol 1 (that uses TOTP) is used to transfer the encrypted TOTP from PC to the smartphone, and the client plays a role of another channel from the smartphone to PC by entering the decrypted OTP into PC. Here, both channels (from PC to the smartphone and vice versa) can be replaced with other channels such as Bluetooth, and the whole authentication procedure can be automated. This will significantly enhance the usability of the authentication protocol. However, in another aspect, not all PCs are equipped with the Bluetooth module. Also, even though PC has the Bluetooth module, it might be an annoying job to execute the pairing whenever the client uses a device that she has never paired before. In that sense, Protocol 1 with visual channel and client's entering PIN are easier to be deployed in the current environment.

V. RELATED WORK

A closely related vein of research is trust establishment for group communication using cognitive capabilities. Examples of such works include SPATE [13], GAnGS [2], and SafeSlinger [5]. None of these works use visualization as reported in this work, although they provide primitives for authentication clients and establishing trust.

Another closely related work is "Seeing-is-Believing" (SiB) [14],[15], which uses visual channels of 2D barcodes to resist the man-in-the-middle attack in device pairing. Though it utilizes similar tools by using the 2D barcodes for information representation, and the visual channel for communicating this information, these protocols are further more generic than those proposed in [14]. These protocols are tailored to the problem settings in hand, e-banking, with a different trust and attack model than that used in [14] which results into different guarantees as explained earlier in this paper. To prevent against phishing, Parno et al. suggested the use of trusted devices to perform mutual authentication and eliminate reliance on perfect client behavior.

VI. CONCLUSION

The two authentication protocols are proposed to show how visualization can enhance usability and security. Moreover, these two protocols helps to overcome many challenging attacks such as keylogging and other malware attacks. This system can be

©IJRASET 2015: All Rights are Reserved

implemented in many real world applications since it utilizes simple technologies and feasible to use as android application.

REFERENCES

[1] BS ISO/IEC 18004:2006. Information Technology. Automatic Identification and Data Capture Techniques. ISO/IEC, 2006.

[2] D. Boneh and X. Boyen. Short signatures without random oracles. In Proc. of EUROCRYPT, pages 56–73, 2004.

[3] C.-H. O. Chen, C.-W. Chen, C. Kuo, Y.-H. Lai, J. M. McCune, A. Studer, A. Perrig, B.-Y. Yang, and T.-C. Wu. Gangs: gather, authenticate 'n group securely. In J. J. Garcia-Luna-Aceves, R. Sivakumar, and P. Steenkiste, editors, MOBICOM, pages 92–103. ACM, 2008.

[4] N. Doraswamy and D. Harkins. IPSec: the new security standard for the Internet, intranets, and virtual private networks. Prentice Hall, 2003.

[5] M. Farb, M. Burman, G. Chandok, J. McCune, and A. Perrig. Safeslinger: An easy-to-use and secure approach for human trust establishment. Technical report, CMU, 2011.

[6] H. Gao, X. Guo, X. Chen, L. Wang, and X. Liu. Yagp: Yet another graphical password strategy. In Proc. of ACM ACSAC, pages 121– 129,2008.

[7] S. Goldwasser, S. Micali, and R. L. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. SIAM Journal, 1988.

[8] S. Goldwasser, S. Micali, and R. L. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. SIAM Journal, 1988.

[9] E. Hayashi, R. Dhamija, N. Christin, and A. Perrig. Use your illusion: secure authentication usable anywhere. In Proc. of ACM SOUPS, 2008.

[10] A. Hiltgen, T. Kramp, and T. Weigold. Secure internet banking authentication. IEEE Security and Privacy, 4:21–29, March 2006.

[11] N. Hopper and M. Blum. Secure human identification protocols. In Proc. of ASIACRYPT, 2001.

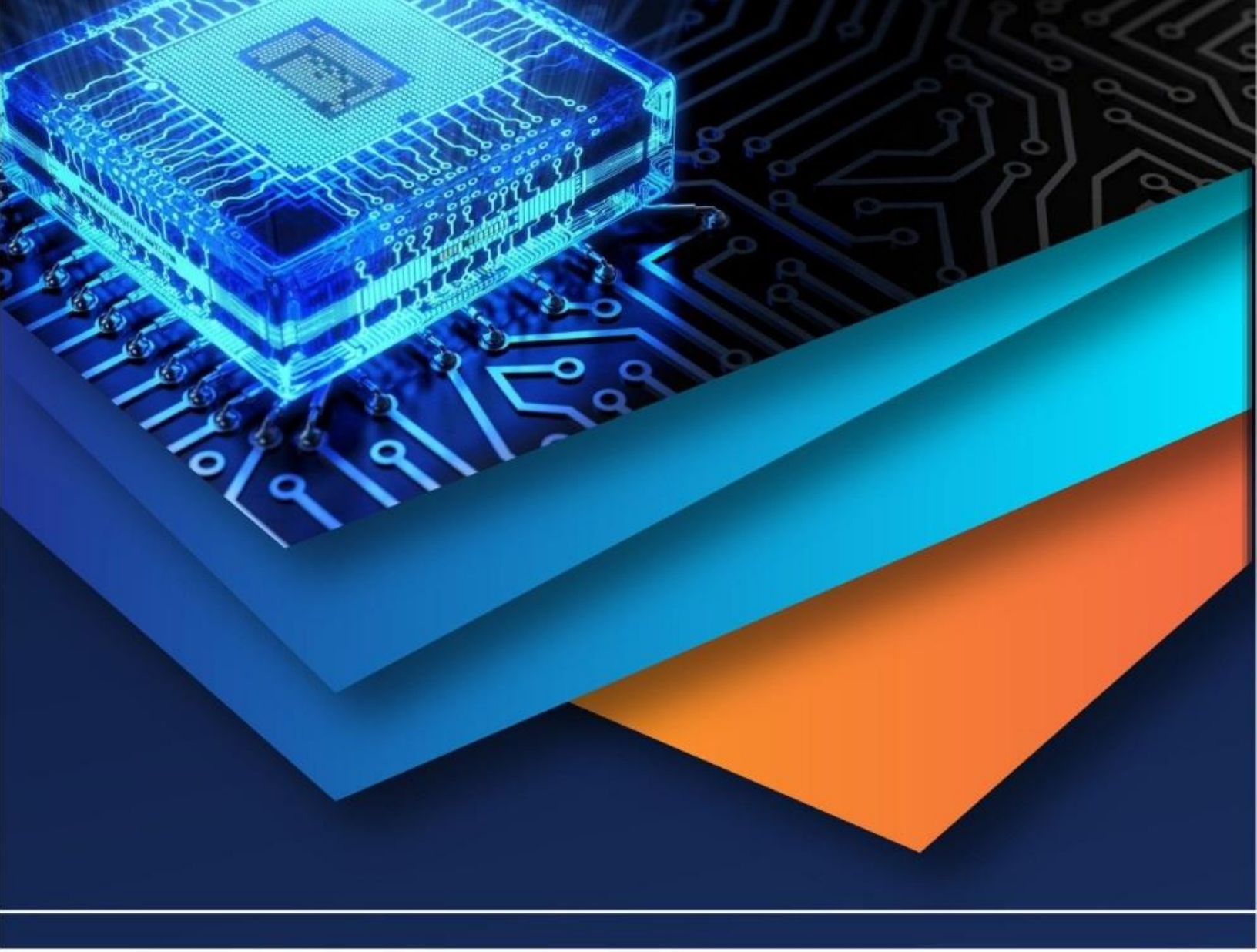
[12] J. Katz and Y. Lindell. Introduction to modern cryptography. CRC Press, 2008.

[13] M. Kumar, T. Garfinkel, D. Boneh, and T. Winograd. Reducing shoulder surfing by using gaze-based password entry. In Proc. of ACM SOUPS, pages 13–19, 2007.

[14] Y.-H. Lin, A. Studer, Y.-H. Chen, H.-C. Hsiao, E. L.-H. Kuo, J. M. McCune, K.-H. Wang, M. N. Krohn, A. Perrig, B.-Y. Yang, H.-M. Sun, P.-L. Lin, and J. Lee. Spate: Small-group pki-less authenticated trust establishment. IEEE Trans. Mob. Comput., 9(12):1666–1681, 2010.

[15] J. M. McCune, A. Perrig, and M. K. Reiter. Seeing-is-believing: Using camera phones for human-verifiable authentication. In Proc. of IEEE Symposium on Security and Privacy, pages 110–124, 2005.

©IJRASET 2015: All Rights are Reserved



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)