Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/cose

Computers & Security

Transaction authentication using complementary colors☆

YoungJae Maeng [a,b], Aziz Mohaisen [c,1], Mun-Kyu Lee [a], DaeHun Nyang [a,*]

[a] Department of Computer and Information Engineering, INHA University, YongHyun-dong, Nam-gu, Incheon 402-751, Republic of Korea

[b] The Attached Institute of ETRI, Daejeon 305-390, Republic of Korea

[c] Verisign Labs, USA

ARTICLE INFO

Article history:
Received 5 November 2013
Received in revised form
21 September 2014
Accepted 5 October 2014
Available online 24 October 2014

Keywords:
Visual cryptography
Authentication
Complementary colors
Subtractive color mixing
Additive color mixing

ABSTRACT

In this paper, we introduce a transaction authentication solution that provides compatibility with any banking transactions. Our solution is based on a novel visual cryptographic scheme that supports multiple uses of a single static share, unlike existing techniques in the literature of visual cryptography. To support multiple uses, we utilize complementary colors in the subtractive color mixing model. In our solution, a user is asked to overlap an issued visual cryptography card on an encrypted image that is rendered on a screen to check transaction information and transaction authentication number. We analyze the security of our scheme against various attacks and show its effectiveness using a usability study.

1. Introduction

In banking systems, computer terminals are the main player in performing sensitive financial transactions. Such a role played by terminals shifted the attack surface and defense mechanisms from the network to include end hosts over a number of years in continuous developments. Historically, many Internet banking services and solutions were designed to protect users only against network-based attacks utilizing various transport layer security mechanisms, and overlooking the potential attacks from within the terminals. Those mechanisms have been proven to be limited in many ways, mainly because of the rise of malware attacks, in which terminals are infected by malicious pieces of software that have given the adversary a full control over the terminals, including the banking systems running on top of them.

Among many kinds of attacks that malware can mount against Internet banking services is a client-side transaction-manipulation attack in which the adversary controls a user's established session and abuses secret information (e.g. password, PIN, or OTP string) entered by the user for completing a manipulated transaction (Oppliger et al., 2009). Because early Internet banking solutions did not ensure that the present

☆ This research was supported in part by Inha University, in part by the MSIP, Korea, under the ITRC support program [No.1711013961] supervised by the NIPA, and in part by the IT R&D program of MOTIE/KEIT [No.10039180].

* *Corresponding author.*
E-mail address: nyang@inha.ac.kr (D. Nyang).

[1] The views and opinions expressed in this paper are the views of the author, and do not necessarily represent the policy or position of Verisign, Inc.

http://dx.doi.org/10.1016/j.cose.2014.10.001
0167-4048/

transaction is what the user has intended, they were vulnerable to the attack. In this attack, the adversary aims to trick the user into completing a manipulated transaction by displaying an uncompromised transaction information at the client side. Accordingly, the user inputs the secret information required for completing the transaction, making the adversary capable of controlling the session established by the user. Refer to Fig. 1 for an illustration of the attack.

In retrospect, to ensure a level of defense against such attacks and to avoid liability, online banking providers initially urged users to adequately protect their devices by following best practices. However, given the ever-increasing levels of sophistication of today's security landscape, which may include zero-day vulnerabilities and attacks, online banking providers moved towards a more active role by equipping users with trusted devices to secure their banking services from potential attacks at the terminal. For that, some banks assume that the user terminal is controlled by the attacker; hence, users are provided with additional devices with a display that serves as a trusted endpoint in the communication between customer and bank server. Solutions that aim at preventing or mitigating it include ZTIC (Weigold et al., 2008), CAP (Drimer et al., 2009), Transaction Signing (Hiltgen et al., 2006), and QR-TAN (Starnberger et al., 2009), which introduce additional trusted devices for transaction authentication. Some of them, such as ZTIC, have already been certified in many countries and deployed by some banks (IBM (ZTIC), 2014; State-of-the-art technolo, 2014). This trusted device is then used to either display transaction information (in ZTIC and QR-TAN) or generate transaction-dependent responses by asking the user to input part of the transaction information (in CAP). While such hardware-based solutions prevent the attack, that advantage comes at an extra cost with the additional devices. In addition, users in CAP (Drimer et al., 2009) are asked to input redundant transaction information more than once. Moreover, solutions like ZTIC (Weigold et al., 2008) and Transaction Signing (Hiltgen et al., 2006) that have a standard USB (Universal Serial Bus) connector cannot be used in other devices such as a mobile phone.

In this paper, we introduce a transaction-authentication scheme that has the following properties:

1. Low-cost: A credential is a sheet of transparent paper.
2. Easy to issue a credential: Credential issuing process is just to print secret color patterns on a transparent paper.
3. No input redundancy: It does not require a user to input transaction information multiple times.
4. Universal usage: A standard screen and an input device are enough to use the credential.

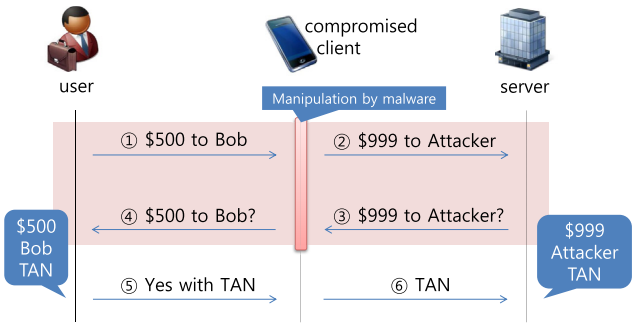


Fig. 1 – Client-side transaction-manipulation attack (TAN: Transaction Authentication Number).

To achieve these goals, we propose a novel visual cryptographic scheme, and utilize it for authentication without trusted modules or devices. The proposed scheme takes advantage of the interesting interaction between subtractive and additive color mixing models, where the subtractive mixing is a model for printing on paper and the additive mixing for light on screen. It is enough for a user to overlap a pre-issued transparent paper printed with a color pattern over a screen with a random pattern when completing the transaction. This overlapping action causes the interaction between two color mixing models and shows a secret message that will be used for transaction authentication. Our contribution stated above can be viewed as an unconventional trusted device that banking services can provide to users for secure transactions, while providing all the operational advantages highlighted earlier.

The organization of the rest of this paper is as follows. In Section 2, we explain a new visual cryptographic scheme which utilizes complementary colors in color mixing models. In Section 3, we introduce the main protocol of our proposal. In Section 4, we show our implementation and evaluation results in terms of usability. In Section 5, we analyze the security of our proposal. In Section 6, we review prior works on visual secret sharing. Finally, concluding remarks and future research directions are drawn in Section 7.

2. Exploiting color models for visual encryption

In this section, we explain two color mixing models that play significant roles in our scheme: the subtractive and the additive. In the subtractive color mixing model that has three primary colors (cyan, magenta and yellow; CMY for short), the more colors are mixed, the darker the resulting color becomes. This model is applied to create a variety of colors when printing on a paper and in painting. In the additive color mixing model, where the three primary colors are red, green and blue (RGB for short), the more colors are mixed, the lighter the consequent color becomes. This model is used for television and computer monitors. In the subtractive and the additive color models, the complementary colors of red, green, and blue are cyan, magenta, and yellow, respectively. It is well known that when subtractively mixed, pairs of complementary colors can be used to produce the black (K) color (Color wheel, 2014).

We observed an interesting interaction of the two coloring systems: when a color printed on a transparent paper (the subtractive model) is overlapped over the complementary color on a screen with a backlight (the additive model), the resulting color is still dark gray, although not completely black. Fig. 2 shows this observation—the dark gray color is denoted as K throughout the paper. On the other hand, even if one of the RGB is mixed with one of CMY that is not a complementary color, the RGB color does not lose its original color.

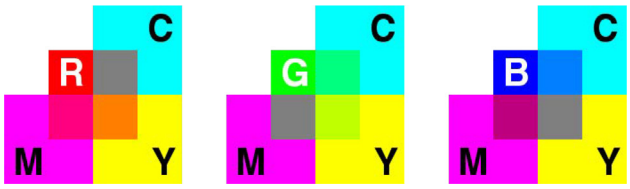


Fig. 2 – Overlaying RGB colors on a transparent paper over CMY colors on a screen results in gray [best viewed in colors].

For example, R becomes K when mixed with C, which is complementary to R, but otherwise remains almost R, absorbing M or Y.

Using the complementary color interaction between an RGB layer and a CMY layer, we can make a visual encryption scheme, where a user is issued as a key a transparent card filled with a random color pattern, and the user is required to overlap the card over the other layer displayed on a screen to recognize the secret information. We remark that when we mix two color systems, displaying CMY on a screen overlapped with an RGB card generates better results than vice versa, according to our experimental results. See Fig. 3 for an example. We note that there are two differences in this approach from traditional visual cryptography. First, our approach is based on creating a new color by overlapping two colors, while all existing proposals in visual cryptography are based on density control of a color as shown in Fig. 4. Second, we use additive and subtractive color systems simultaneously. As shown in Fig. 3, we observed that a mixture of the two color systems is dominated by the subtractive color system, although it does not follow exactly the subtractive model. Consequently, a dark gray is shown as a mixture of complementary colors instead of black. We utilize this basic idea to create a secure transaction authentication protocol.

3. Our proposal

In this section, we introduce the system and attacker models followed by our main protocol proposal.

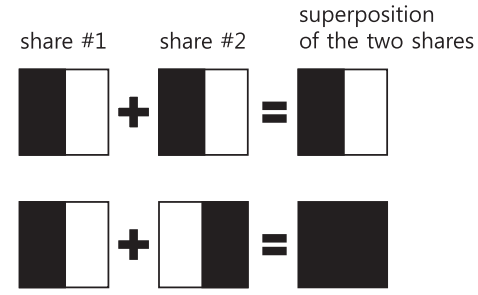


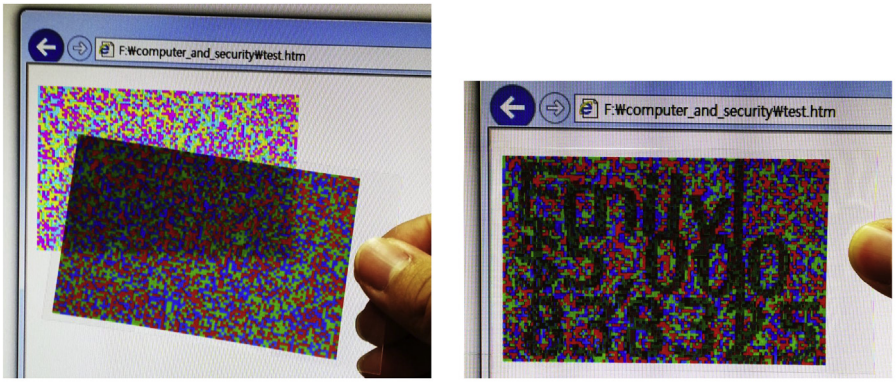
Fig. 4 – Traditional Visual Cryptography: Overlaying secret shares to increase density with superposition.

3.1. *System model*

Our system model consists of four different entities (or participants), which are a user, a visual cryptography card (VCC), a user's terminal (client), and the server. The user is an ordinary human, limited by all human's limitations and shortcomings, including limited capabilities of performing complex computations or remembering sophisticated cryptographic keys. With a user's terminal such as a personal computer, or a smartphone, the user can log in a server of a financial institute (bank) for financial transactions. After a successful login by proving possession of a valid secret information, the user can do financial transactions such as money transfer and bill payment. A VCC is a transparent card that contains a secret key used for authenticating the transaction. Finally, the server is the last system entity which belongs to the financial institute and performs the back-end operations by interacting with the user (terminal and VCC) on behalf of the bank to enable the authentication process.

3.2. *Attacker model*

The main attacker model of our work is a client-side transaction-manipulation attack. We assume that the adversary has a malware in the user's terminal and privileges to control any resources of the client. For example, it can log keystrokes and access a private key file and memory, and it



(a) Before overlapping a transparent RGB card on screen (b) When a card is overlapped on screen

Fig. 3 – Overlapping an RGB card over the CMY layer on a screen enables a user to recognize the secret.

can even capture the screen. However, the attacker does not have access to a VCC. The goal of the adversary in our attacker model is either to reconstruct the VCC by analyzing information transferred between the server and the user, or to find a way to manipulate the transaction without restoring the VCC.

3.3. *Issue of VCC*

A user is issued with a VCC when she signs up the banking service, and the color pattern is stored in the server's database. To issue a VCC, three candidate VCCs ($V1, V2, V3$) are generated.

We denote $\mathbb{S}_A$ as a set of RGB colors ($\mathbb{S}_A = \{R, G, B\}$), and $\mathbb{S}_S$ as a set of CMY colors ($\mathbb{S}_S = \{C, M, Y\}$). Assume that a VCC is composed of $h \times w$ blocks, where a block is defined as a unit square to which a single color is assigned.

1. For each of $h \times w$ positions (x, y), the server assigns a random permutation of (R, G, B) to $(V1_{x,y}, V2_{x,y}, V3_{x,y})$, where $V1_{x,y}$ is a block in (x, y) position of V1.
2. $V1, V2$ and $V3$ are stored in the database.
3. V1 is issued to a user.

We will discuss the rationale for using three VCCs instead of one in Section 5.5.

3.4. *Transaction authentication protocol with VCC*

We extend the basic idea of the visual encryption scheme in Section 2 to transaction authentication. A user performing a financial transaction executes the following protocol:

1. **Transfer request on the user (client) side.** After signing-in on a banking service, the user moves to 'make transfer' menu, and inputs transfer details such as source account, transfer amount, and recipient's account information. The client requests a transaction challenge of the server by sending the transfer details.
2. **Response with transaction challenge on the server side.** After receiving the transfer details, the server checks the user's balance. If there is enough balance, the server generates a Transaction Authentication Number (TAN), and constructs a transaction challenge (TC). TC contains *TAN*, recipient's name, transfer amount, and a bar— we will describe the construction of the TC in section 3.5 in more detail. The server sends the TC to the client.
3. **Transfer information check and *TAN* input at the user side.** The client receives the TC and displays it on the screen. To check the transfer information (the recipient's name and the transfer amount) and the *TAN*, the user overlaps the VCC on the TC rendered on the screen of the client as in Figs. 3 and 5. The user recognizes the resulting characters of the transfer information, the *TAN*, and a bar (this will be addressed in Section 5.2). The user inputs the *TAN* if there is only 1 bar on the resulting image and the transfer information is correct. The client then forwards the *TAN* to the server.
4. **Verification of *TAN* on the server side.** The server confirms the transfer only if *TAN* that is received matches the one that it has sent.

3.5. *Construction of TC*

TC construction method is the most important part in our protocol, because the adversary can obtain many TC's from the compromised client and try to analyze those TC's to reconstruct the user's VCC.

To construct a TC, the server prepares a bar, the *TAN* and the transaction information including recipient's name (*RN*) and transfer amount (*TA*). Either numbers or alphanumeric characters can be chosen for a *TAN*, depending on input method available (10-key keypad or qwerty keyboard), user convenience, or the required security level. The bar is used to

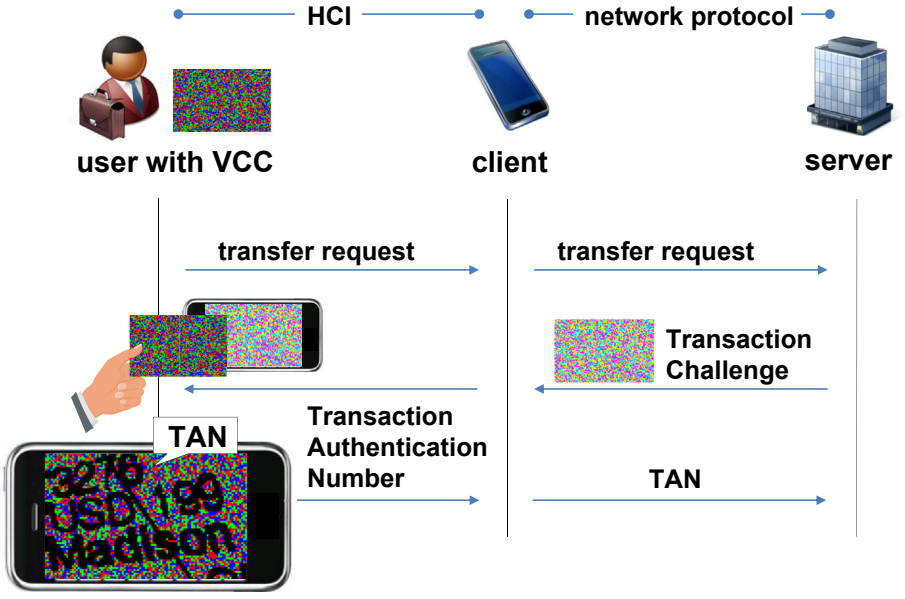


Fig. 5 – Transaction authentication protocol using complementary colors.

make the user confirm that the TC has not been manipulated by a cut-and-paste attack, which will be discussed in Section 5.2. We denote the set of blocks occupied by the bar, *RN*, *TA*, and the *TAN* as *TS* as shown in Fig. 6. The size of the TC is the same as the size of the VCC.

1. The server retrieves the user's $V1, V2$ and $V3$ from the database.
2. The server prepares images for *TS* as follows. For *RN*, *TA*, and *TAN*, the font size is chosen to be 20 color blocks in height. The lengths of those three strings are variable. A TC is composed of three horizontally-divided regions, called R_1, R_2, and R_3. The server randomly assigns *RN*, *TA*, and *TAN* into three regions. For each region, the server chooses a set of parameters, (x, y, θ), where x and y are the coordinates of the top left corner of the string, and θ determines how much the string will be rotated. x is randomly determined by considering the width of a string not to go beyond the edge of a TC, and y is randomly chosen such that the string should fit into a region with a variation of ± 3 color blocks. θ ranges from -10° to 10°. The three strings are rotated and laid out properly so that they do not overlap. The bar is drawn from $(x_1, 0)$ to $(x_2, h-1)$, where x_1 and x_2 $(0 \le x_1, x_2 \le w-1)$ are randomly chosen, and the thickness of the bar is three color blocks. The number of blocks constituting *TS* should be approximately 1/3 of $h \times w$. Refer to Fig. 6 for sample images. The server also generates *TS′* and *TS″*, two sets of fake transaction information such that each set occupies approximately $wh/3$ blocks. *TS*, *TS′* and *TS″* should be disjoint from each other.
3. The server constructs a TC as follows. For each block of an empty TC of size $h \times w$, if the block belongs to the region where *TS* occupies, the server assigns the complementary color of the color on the corresponding block of $V1$; if the block belongs to *TS′*, it fills the block with the complementary color on the corresponding block of $V2$; otherwise, it fills it with that of $V3$. An example of the construction is shown in Fig. 7.

Whenever a user requests a transaction authentication, the server generates a fresh TC by running the above algorithm to frustrate various types of attacks which will be discussed in Section 5.

4. Implementation and evaluation

We conducted a study to find how much the difference is between the authentication time of our proposal and that of the random 6-digit input. For the study, we used a VCC printed on a normal OHP film with pure RGB colors using Samsung CLP-705ND color laser printer. The size of the VCC is the same as that of a typical credit card ($8.5 \times 5.4\ cm^2$), and the size of each color block is $0.79 \times 0.79\ mm^2$. With $w = 107$ color blocks in width and $h = 68$ color blocks in height printed on the card, the printed area is $8.45 \times 5.37\ cm^2$. To represent three strings for *TAN*, a recipient name and the amount of transfer, we used 'verdana' font with a size of 20 points. While any number or alphanumeric characters can be used for a *TAN*, we chose a 6-digit number.

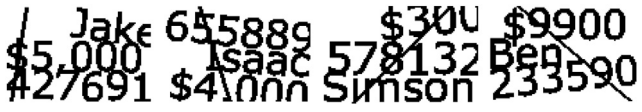


Fig. 6 – Examples of *TS*.

4.1. User study design

Our proposal using 6-digit number *TAN* was evaluated against the 6-digit OTP (One Time Password) using a repeated measures within-participants design. To verify how the proposal works in practice, we asked subjects to use their own computer monitors, of which color settings varied. Because of the difference in the color settings of users' own monitors, the TC and the image overlapped with the VCC might not look identical in their colors.

4.2. Participants

To evaluate the usability of our proposal, we conducted a user study with 20 subjects, who were recruited from our local college. They received a coffee coupon in lieu of their participation. Their actual ages ranged between 21 and 34 years, with an average age of 26.05 years.

4.3. Procedure

The study was conducted in an isolated room of our college. Subjects were given a VCC, and then they were briefed on how to calibrate the size of a TC on their monitors and to perform a transaction with the VCC. The calibration was to adjust the size of the TC on a monitor to match that of VCC using arrow keys. The adjusted size was saved in the server and was retrieved to be used later on.

After calibrating the VCC, a subject was requested to run our protocol by (1) finding a TC on a monitor, (2) overlapping the VCC with the TC on the monitor, and (3) typing the *TAN* rendered from the overlapped image.

After one practical trial each for calibration and for authentication as described above, the participants were requested to run one calibration trial and ten transaction authentication trials. Calibration trials were for measuring the elapsed time for calibration. Transaction authentication trials were to measure the successful authentication probability and time. For comparison, participants were asked to input a random 6-digit number shown in their screens ten times.

4.4. Results

Fig. 8 shows the elapsed time for calibration. We observed that the average elapsed time for calibration was 42.2 s.

Fig. 9 shows comparatively the elapsed time of our proposal and that of a random 6-digit input. The elapsed time of our proposal includes time for overlapping images, and recognizing and inputting the *TAN*. We found that the average elapsed time of our proposal is longer than that of the random input by 11.08 s. The average success probability of 6-digit input was 99%, whereas that of ours was 93.5%.

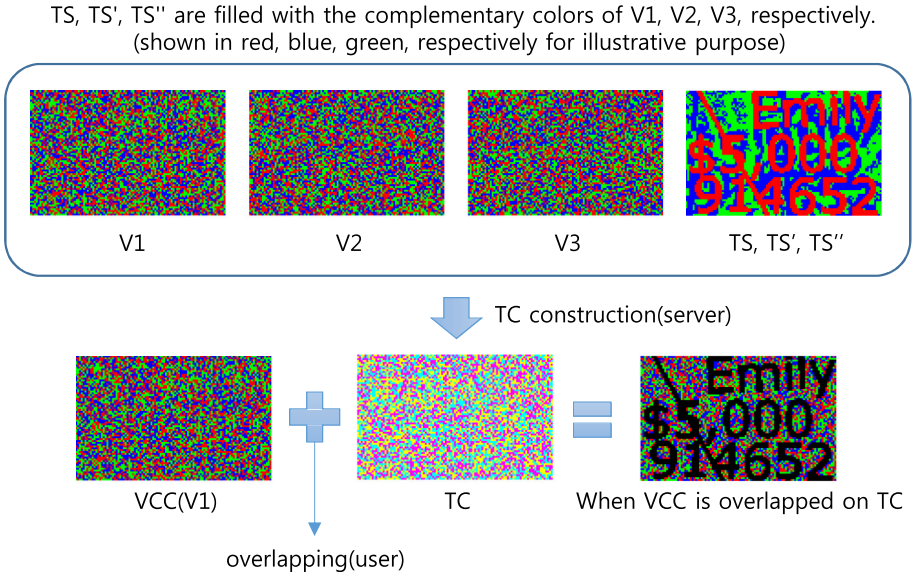


Fig. 7 – TC construction and decryption of *TS*.

4.4.1. *Remark on durability*

We remark that a VCC might be worn out as time goes on, because it is repeatedly used during a long period. To check the durability of the card, we made a VCC with a normal OHP film using Samsung CLP-705ND color laser printer. The VCC was tested 300 times in a typical environment, where a user stored the card in his/her wallet and pulled out for every authentication session. Even after 300 trials, the card remained without any serious flaw as shown in Fig. 10, and we could successfully perform authentication with the card. We also remark that another layer of protection material may be applied to enhance the durability for deployment.

5. Security analysis

5.1. *Basic attacks*

In this section, we analyze the security of our scheme against basic attacks such as random guessing, brute-force attack, key-logging attack, and client-side transaction-manipulation attack. We also compare our scheme with legacy solutions such as 6-digit PIN, matrix card (grid card), and OTP (6-digit) in terms of these attacks. See Fig. 11. In case of the matrix card in Fig. 11(b), given '3F' as a challenge, the user must answer with 'H7' on an intersection cell located in '3' row and 'F' column. This procedure would be repeated with different challenges to satisfy the required security level. For a fair comparison, we consider a 10-by-10 matrix card with 100 two-digit decimal numbers.

5.1.1. *Random guessing*

An attacker may pass the authentication protocol by randomly guessing the *TAN*. The success probability of this guessing depends on the size of the search space of a *TAN*. For example, for a 6-digit *TAN*, it is $1/10^6$, which is the same as the legacy 6-digit PIN. The other legacy solutions of which response is 6 digits long have the same level of security.

5.1.2. *Brute-force attack*

An adversary may try to reconstruct a VCC using a brute-force attack. The attacker is assumed not to have a priori knowledge of the VCC, and to perform random guessing on the colors of VCC blocks. Since the probability of correctly guessing one block in the wild is 1/3, the probability of recovering the VCC is $(1/3)^{107\times68}$ (more than 11,532 bits of security), which is small

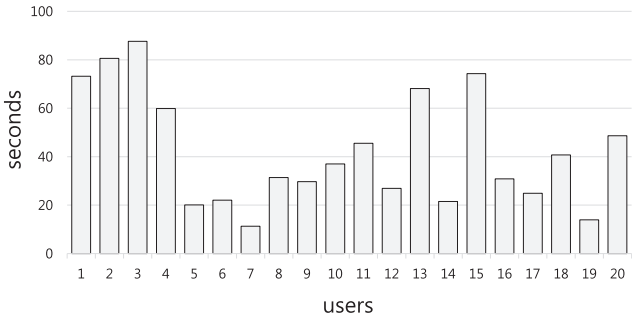


Fig. 8 – Elapsed time for calibration for 20 users.

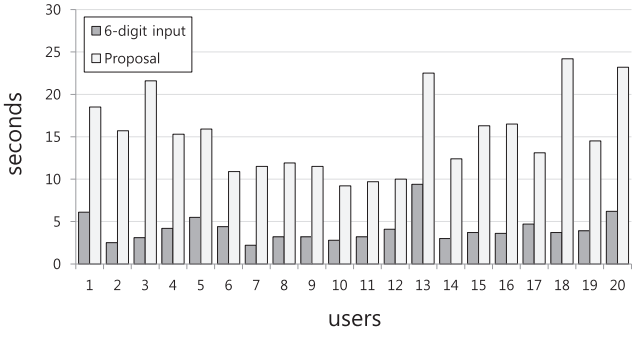
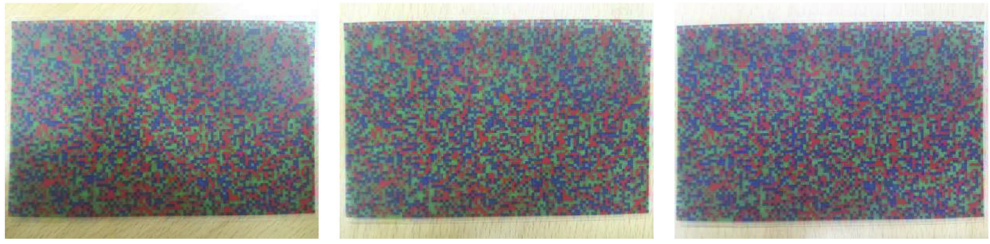


Fig. 9 – Comparison of our proposal to a random 6-digit input (averaged on 10 trials per user).

(a) VCC in a wallet



(b) VCC remained clean after 300 trials.

Fig. 10 – Durability test.

enough to frustrate the brute-force attack. For the matrix card, the probability of recovering the specific card is 1/100!, because there are 100 numbers to be permuted. Assuming that the OTP is implemented using a PRNG (Pseudo-Random Number Generator) with a 128-bit seed, the probability of obtaining the seed is $1/2^{128}$ (negligible).

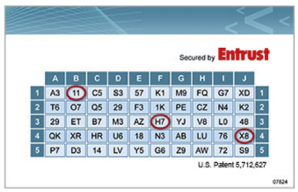
5.1.3. *Key-logging attack*

An attacker may log a challenge from an authentication server and the user's key entry to recover a user's secret information such as PIN, matrix card, or a seed. Our scheme as well as OTP is secure against the key-logging attack, because an expected response is different for every session. However, 6-digit PIN is vulnerable to this attack, because simple replay can pass the test. For the matrix card, an attacker may accumulate the information of logged sessions to increase the probability of authentication success. Assume that a server sends three distinct queries in a single session. When the attacker records only one session, the probability to succeed in the next session is $(3 \times 2 \times 1/100 \times 99 \times 98) \approx 6/10^6$. If two sessions are recorded, the probability is greater than $(97 \times 96 \times 95/100 \times 99 \times 98) \times (6 \times 5 \times 4/100 \times 99 \times 98) \approx 0.00011$. If three sessions are logged, the probability is greater than $(97 \times 96 \times 95/100 \times 99 \times 98) \times (94 \times 93 \times 92/100 \times 99 \times 98) \times (9 \times 8 \times 7/100 \times 99 \times 98) \approx 0.0004$.

5.1.4. *Transaction manipulation attack*

If the protocol does not ensure that the present transaction is what the user has intended, an attacker can make the user complete the transaction with attacker's intention by showing a manipulated page to a user. Unlike OTP, matrix card, and PIN, our scheme can prevent this attack because it has a transaction-dependent challenge. Table 1 summarizes the security of OTP, matrix card, PIN, and the proposed method against the above four attacks. We remark that there are different solutions to prevent a transaction manipulation attack, such as ZTIC, Transaction Signing, QR-TAN, and CAP which were already mentioned in Section 1. The suggested cryptographic algorithms and protocols as well as hardware requirements for these solutions are compared in Table 2. As shown in the table, our scheme is a more cost-effective solution than the other transaction manipulation attack-immune schemes, in the sense that it does not require any additional hardware device and it can be used only with a standard screen and an input device.

(a) OTP with 6 digits



(b) Matrix card [52]

Fig. 11 – Examples of OTP device and matrix card (Entrust IdentityGuard, 2014).

Table 1 – Security comparison of authentication schemes against random guessing (RG), brute force (BF), key-logging (KL), and transaction-manipulation (TM) attacks.

	RG	BF	KL	TM
6-digit PIN	$1/10^6$	$1/10^6$	1	Vulnerable
OTP	$1/10^6$	$1/2^{128}$	N.A.	Vulnerable
Matrix card	$1/10^6$	1/100!	>0.0004[a]	Vulnerable
Proposed	$1/10^6$	$(1/3)^{107\times68}$	N.A.	N.A.

[a] After logging three sessions.

Table 2 – Solutions to prevent transaction manipulation attacks.

	Suggested protocols/algorithms and a dedicated hardware device
ZTIC (Weigold et al., 2008)	SSL/TLS with a dedicated hardware using USB
Transaction Signing (Hiltgen et al., 2006)	Public key encryption with a dedicated hardware using USB
QR-TAN (Starnberger et al., 2009)	Public key encryption with a mobile phone and QR-code
CAP (Drimer et al., 2009)	Compressed version of a MAC with a dedicated hardware and a debit card
Proposed	Visual cryptography with the VCC

5.2. *Cut-and-paste attack*

The purpose of the cut-and-paste attack is to deceive the user with a manipulated TC that contains the recipient name and the transfer amount that has been input by the user and the *TAN* from the adversary's transfer request. A pre-installed malware in a user's client program can mount this attack. When the client sends a transfer request to the server, the malware, which compromises the established session between the client and the server, performs the following procedure to get the manipulated TC as depicted in Fig. 12.

1. Intercepts the TC from the user's transfer session (denoted by TCU),
2. Requests a new session for transfer to the server,
3. Receives the corresponding TC (denoted by TCA),
4. Guesses geometric parameters to represent TS images of TCU,
5. Cuts the recipient's name and transfer amount from the intercepted TCU,
6. Pastes them on the adversary's TCA.

Note that without the user's intervention, the malware cannot decode the *TAN* of TCA because it does not possess the VCC. However, it is hard for the attacker to mount the above cut-and-paste attack, because the adversary should be able to correctly guess random parameters for representation of strings (*RN*, *TA*, and *TAN*), as well as guessing the relative order of strings. For example, a *TAN* may be located at the bottom in the intercepted TC, but at the top in the attacker's TC. Even if the adversary succeeds in guessing those parameters and the order, the cut-and-paste is not easy. A user recognizes easily if she can detect a broken bar instead of a single straight bar.

5.3. *Statistical analysis attack*

An attacker may try to reconstruct the VCC by observing as many TC's as possible. Although the server tries to choose evenly among three colors for every block through many TC's, some area such as the center of a TC cannot but be occupied frequently by TS images, unlike the edge area, during the TC construction. Therefore, a color appears significantly more because a block is located in the central area than the others through multiple TC's, and that implies that the block belongs to a group of blocks consisting of TS images. Unless it belonged to a group, the block should have shown the equivalent frequency of each color given our TC construction strategy. Thus, the server that follows the strategy of allocating the three colors over the card space evenly lacks the color exhausted for the TS images, which is likely to be put in the central area. This inherent property of our proposal might introduce a statistical attack, because the central area has more frequent occurrences of the VCC's complimentary color than the others, while the edge area has far less frequent occurrences of the other colors.

To know whether three colors (CMY) occurred evenly or not, we calculated the standard deviations of three values (the number of occurrences of each color) from the average ($n/3$) for each color block in n TC's. If the standard deviation is large, each color is not evenly used. Fig. 13 shows the visualized standard deviations with different n values (the number of collected TC's). The lighter block means a lower standard deviation throughout a given TC's, which in turn means each of the $V1, V2$, and $V3$ is used evenly, and thus it is hard to guess correctly what color was used in the block.

Using this property, an attacker might be able to mount a statistical analysis attack by collecting as many TCs as possible. The attacker counts the frequency of each color for each block on collected TC's, and chooses the complementary

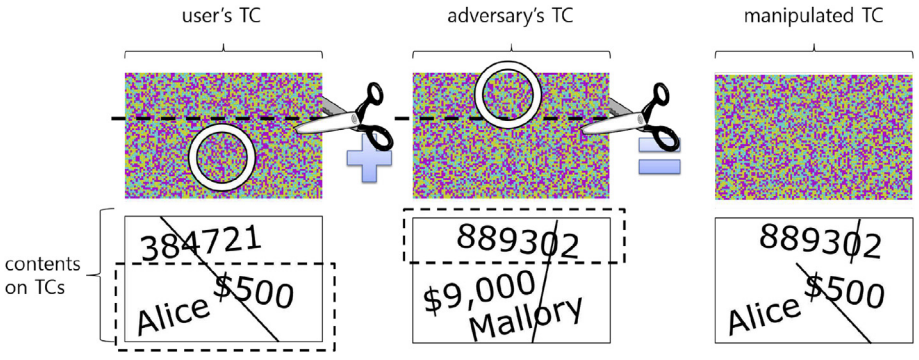


Fig. 12 – An example of cut-and-paste attack.

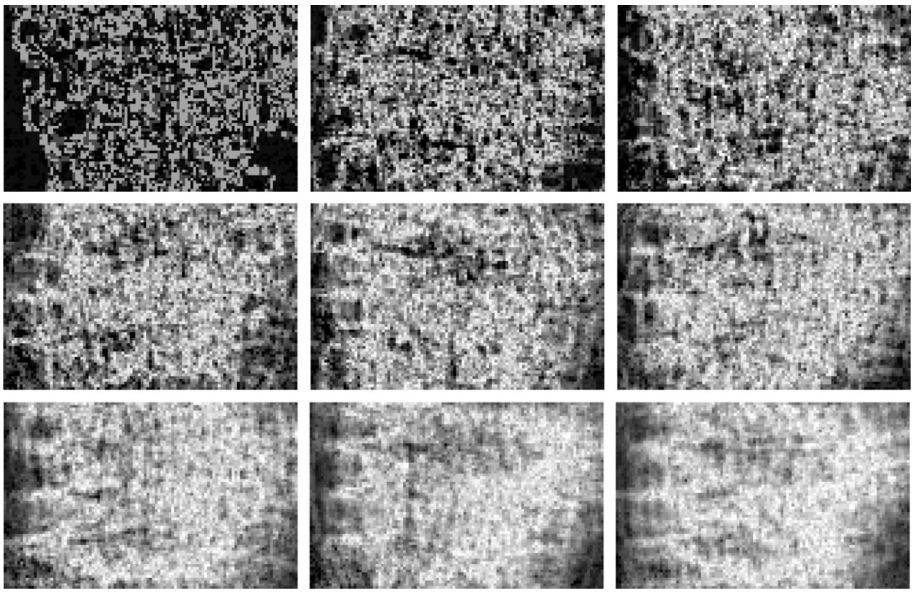


Fig. 13 – Visualized standard deviations of percentages of colors, where $n = 5$, 10, 15, 20, 25, 30, 50, 70, and 90, top-left to bottom-right.

color of the most frequent color for each block to reconstruct the VCC.

Given n TC's, the adversary counts colors represented on each position to mount the attack. Let $x \in [0, w-1]$, $y \in [0, h-1]$, and $k \in [0, n-1]$. Let $V^k_{x,y} \in \mathbb{S}_S$ be a color of a block located in the x-th column and y-th row of k-th TC. Let $E_{x,y}$ be a data structure that consists of ($count_C$;$count_M$$count_Y$), where $count_X$ is the number of occurrences of X (X ∈ {C,M,Y}) at (x, y) of TC's. The adversary can count the colors and then tries to reconstruct the three cards R^0, R^1, and R^2 using Algorithm 1.

In Algorithm 1, the functions MAX, MED, and MIN take $E_{x,y}$ and return a color with the maximum, the median and minimum values in the data structure, respectively. If there is a tie, we set the order as C, M, and Y. The function COMPL takes as an input a color in $\mathbb{S}_S$, and returns its complementary color in $\mathbb{S}_A$—for example, COMPL(C) → R. After running Algorithm 1, $R^0_{x,y}$, $R^1_{x,y}$, and $R^2_{x,y}$ have different colors for each position x, y. The adversary checks the reconstructed cards by overlapping each R on each TC as in Fig. 14. If the adversary can read the *TAN* from the resulting image, the analysis is successful and the adversary can mount transaction-manipulation attack.

```
for y ← 0 to h − 1 do
    for x ← 0 to w − 1 do
        R^0_{x,y} := COMPL(MAX(E_{x,y}))
        R^1_{x,y} := COMPL(MED(E_{x,y}))
        R^2_{x,y} := COMPL(MIN(E_{x,y}))
    end
end
```

Algorithm 1 – An algorithm for statistical analysis.

We implemented the statistical analysis attack by varying the number of acquired TC's. Fig. 14 shows the result of the attack. In the figure, the first, the second, and the third columns represent the resulting images recovered using R^0, R^1 and R^2 with a TC, respectively. R^0, R^1 and R^2 are made after observing $n(= 10, 30, 50, 70, 100)$ TC's. As shown in the figure, the TS in the images were not readable, which implies that it was not possible to recostruct a valid VCC.

Fig. 15 shows the quantitative values for portions recovered correctly. The attacker cannot decide whether a recovered block is correctly guessed or not if it is not black, and therefore, we concentrate only on the black blocks. Moreover, not all the black blocks are correctly-reconstructed ones, because any random pair of complementary colors make a black block. The figure plots the fraction of black blocks that belong to $V1$(VCC) among $h \times w$ blocks in an overlapped image of R^* and a TC. In the plot, a bar represents an average of the correctly-constructed fractions of R^0, R^1 and R^2, and a line over the bar is their standard deviation. The fraction was not increased, even when 500 TC's were analyzed.

The reason that the statistical analysis attack was less successful than expected is due to our TC construction strategy. At the time of TC construction, the server lays a TS image using $V1$, and then it puts a fake TS image on the central area using $V2$. The remaining area is filled using $V3$. Consequently, the color that appears most frequently on a block in the central area might belong to either $V1$ or $V2$, which makes the attack difficult.

5.4. *Statistical attack with aggregated color blocks*

The statistical attack presented in Section 5.3 can be generalized by aggregating color blocks to form a region and by counting the frequency of color patterns at each region. The attack in Section 5.3, thus, can be regarded as a special case of this attack when the number of color blocks of a region is one. We conducted the same experiment with larger regions instead of one color block region, and the results are shown in Fig. 16. The best result was obtained with 2-by-1 and 2-by-2 color block regions, but the advantage of the attacker did not

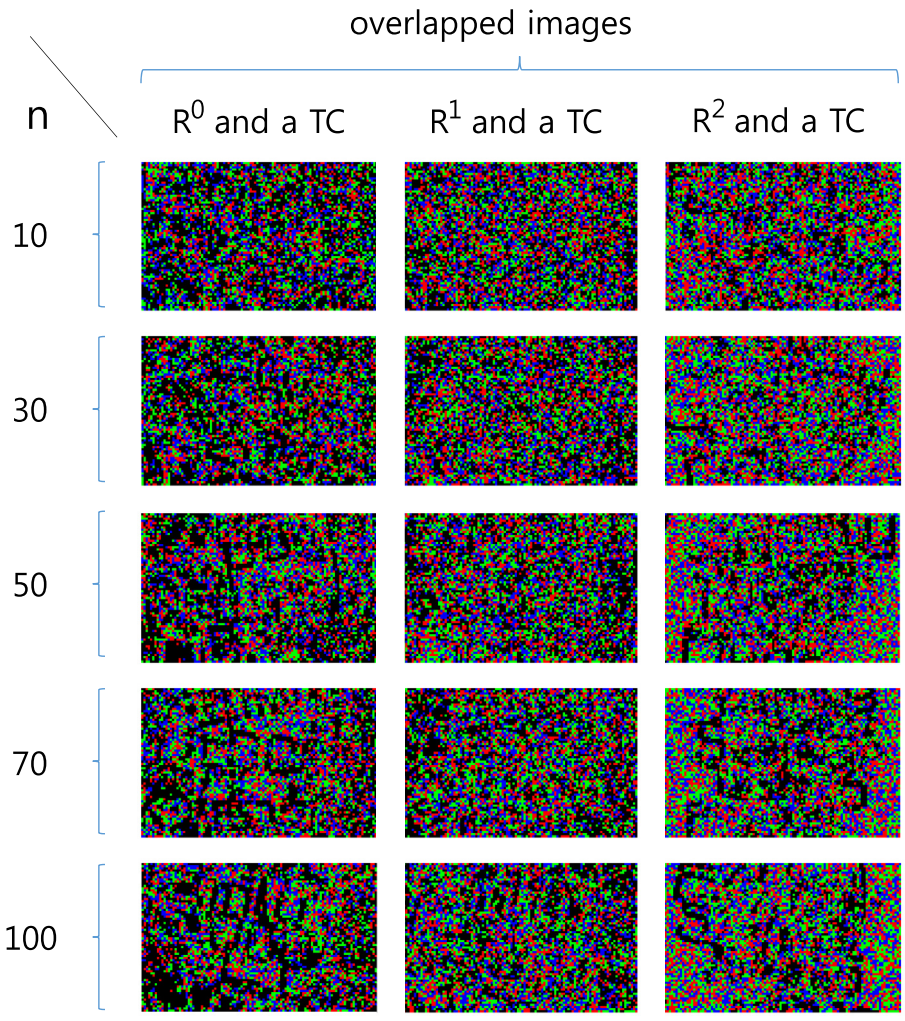


Fig. 14 – Examples of statistical attack with $n = 10, 30, 50, 70, 100$.

significantly increase with this attack. For example, even after the attacker collected and analyzed 500 TC's, the fraction of correctly-reconstructed black blocks was only 25.95%. Larger regions such as 4-by-4, 6-by-6, and 8-by-8 blocks did not produce better results.

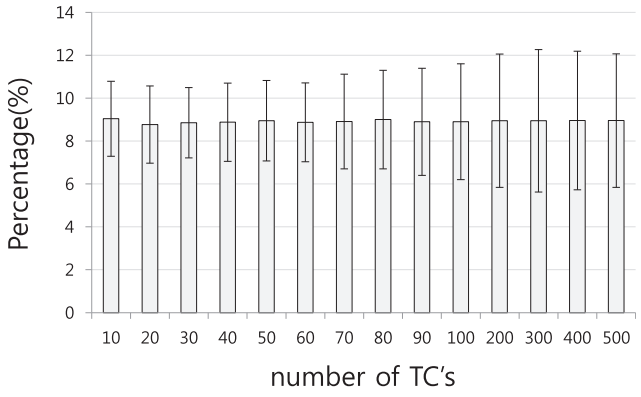


Fig. 15 – Percentages of correctly-reconstructed black blocks by statistical attack.

5.4.1. *Remark on the relationship with stream ciphers*

We note that our scheme can be viewed as a stream cipher (by overlapping colors instead of XORing bits) with a fixed secret key (a VCC card). From that point of view, we can consider two

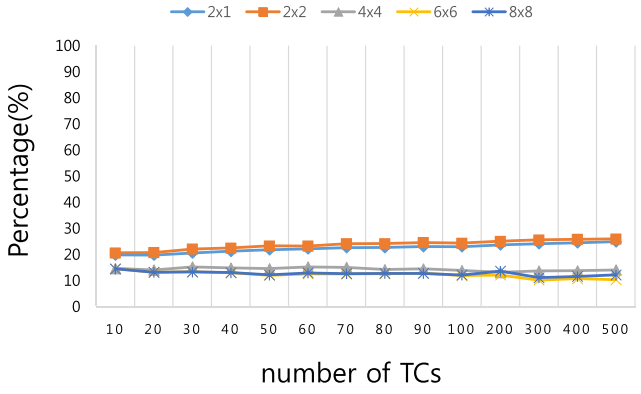


Fig. 16 – Percentages of correctly-reconstructed black blocks with larger regions of 2-by-1, 2-by-2, 4-by-4, 6-by-6, and 8-by-8.

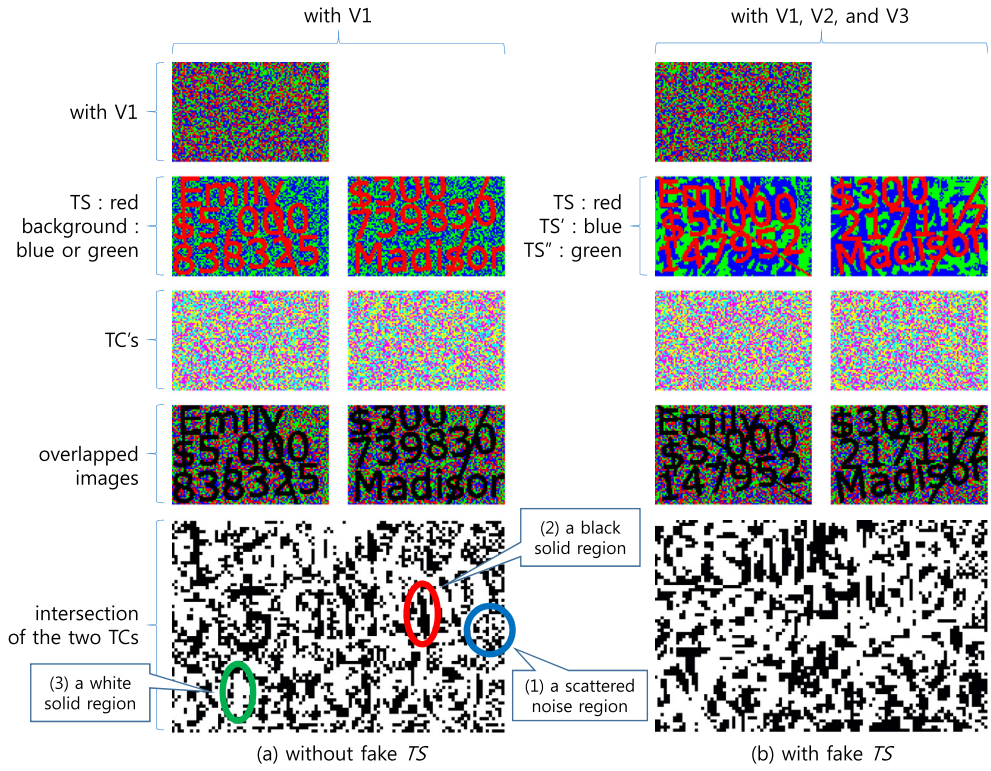


Fig. 17 – Effect of adopting fake TS.

security threats: known/chosen-plaintext-attack (KPA/CPA) and an attack using statistical structure in plaintexts. First, we note that the attacker can only mount ciphertext-only-attack (COA), but it cannot execute KPA/CPA. That is because the deciphered plaintext that is necessary to mount KPA/CPA does not exist in the form of a file or in any digitalized format accessible by the attacker, but it is shown only to the user's eyes in the form of a picture. By capturing the user's input of *TAN*, the attacker might be able to get some information about the VCC. However, the exact positions of the color blocks constituting the *TAN* should be known to the attacker to extract useful information for KPA/CPA, which are hard to be acquired in our scheme because of the *TAN* position randomization. As shown in Fig. 18(b), even one block shift of a TC completely distorts the whole image. Second, considering that TC is a ciphertext and VCC is a key, we can regard the repeated use of a same VCC as repetition of a key sequence of a stream cipher. The period occurring from the repetition can be used to perform the statistical analysis similar to Kasiski's method for Vigenere cipher (Kasiski). This attack is essentially the same as those in Section 5.3.

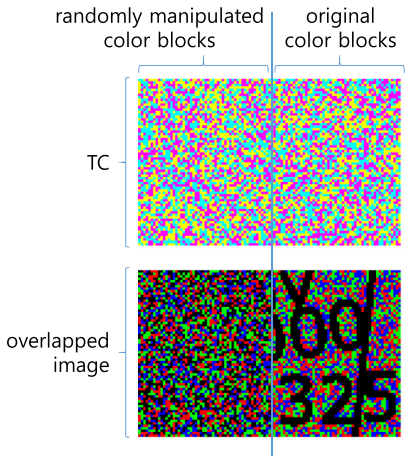
5.5. *Intersection attack*

An attacker may try to extract a piece of information by comparing only two TC's instead of accumulating many TC's. When there is a region that is common to two TC's, the attacker can determine that the region belongs to both TS's of two TC's. The larger the region is, the more likely the region belongs to a TS. See Fig. 17 for details, where Fig. 17(a) shows the case when we do not adopt the fake TS′ and TS″ to the construction of TC, while Fig. 17(b) shows the opposite case. In the intersection result of two TC's shown in the bottom of Fig. 17, a black block is for the same colored blocks and a white block for the different-colored blocks. In the result, there are three types of regions: a black solid region, a white solid region and a scattered noise region. If two TC's are composed of random color blocks, the common blocks represented in a black block should be randomly placed as many scattered noise regions as shown in region (1) of Fig. 17(a). However, there are many black solid regions as well as scattered noise regions, because TS's of two TC's were overlapped each other. For example, the black solid region pointed in region (2) of the figure belongs to the overlapped region of "0" of TC1 and "3" of TC2. The white solid region is generated when the TS (constructed using V1) in TC1 is overlapped with random noises (constructed using V2 or V3) in TC2, and vice versa. For example, the white solid region (3) is made by "M" of TC2 and noises of TC1.

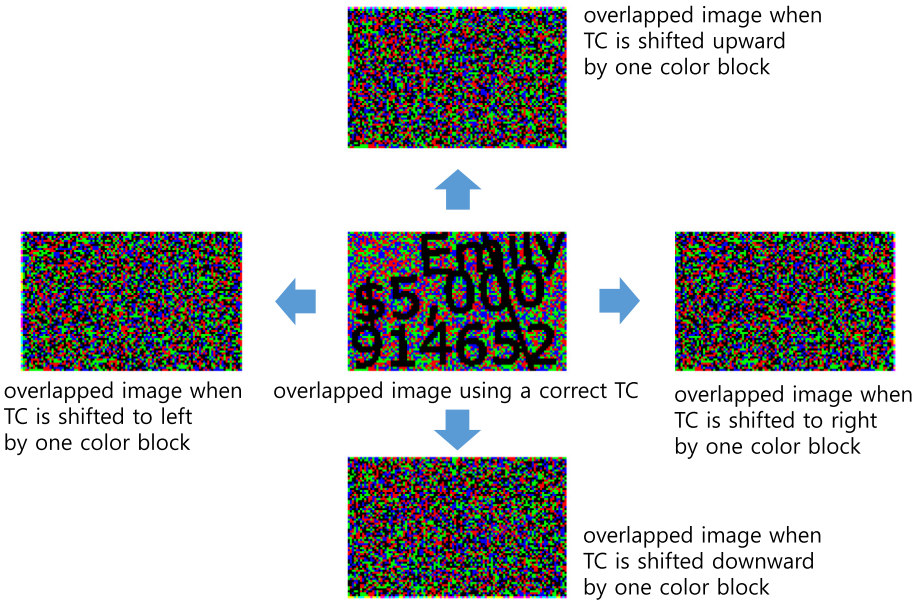
For each case, the attacker can apply a different attack strategy:

1. For the black solid region, it chooses the common colors belonged to the region.
2. For the white solid region, it chooses the color either from TC1 or from TC2. In either case, the probability of guessing correctly is 1/2.
3. For each white block in a scattered noise region, it chooses the remaining color that has not appeared in both TC's. A black block in the region is discarded.

Finally, the attacker constructs a VCC by taking the complementary of the color of each block.



(a) Adding random color noise to TC



(b) Shifting TC by one block

Fig. 18 – Adaptive attack.

However, when we introduce the fake TS′ and TS″ using $V2, V3$, respectively, the first strategy is not applicable because each black solid region can belong to either $V1, V2$ or $V3$. The same difficulty exists in the second strategy, because the overlapped portion of TS and TS′ or TS″ will make the white solid region. Moreover, the scattered noise region used for the third strategy cannot be found at all as shown in Fig. 17(b). Therefore, the intersection attack is not applicable to our proposal.

5.6. *Adaptive attack*

We can also consider adaptive attack models as follows. First, an adative attacker who uses a user as an oracle may try to alter valid TC's adaptively to recover the VCC. However, it is not possible to alter TC's in a meaningful way without the knowledge of the VCC. Thus, the user would not input the *TAN* because the overlapped image does not look regular. As a result, the attacker cannot obtain any information on the VCC. For example, adding a random color noise to TC gives a user only a mangled information. Fig. 18(a) shows the resulting screen that a user sees when a random noise-added TC is given to the user. Taking another example, we can consider an attacker who shifts the TC by one pixel. As shown in Fig. 18(b), however, this attack also does not work because the resulting image is distorted.

Another adaptive attacker may use the server as an oracle to get many valid TC's from the server, and recover the VCC. However, our strategy that the position of the transfer information is randomly determined frustrates the attacker. Let us assume that all the strings hidden in a TC are displayed in fixed segments. Then, by sending transfer information, the attacker will get the valid TC from the server. Here, because the strings for *RN* and *TA* are located in fixed segments, the

attacker can infer colors of those segments in VCC taking the complimentary colors. By adaptively changing transfer information, the attacker will get the color of every segment in VCC. However, because of the random positioning, it is hard for an attacker to guess whether a pixel belongs to a meaningful segment or not. Consequently, the attacker cannot determine whether the color of a pixel should be complimented or not to obtain the corresponding color of VCC.

6. Related works

Our visual encryption scheme is related to visual cryptography in the sense that two pieces of visual data should be overlapped to render meaningful information. In this section, we survey the existing visual cryptography schemes (VCSs), and explain why they are not suitable for our purpose.

Visual cryptography is a technique widely used to securely deliver a secret message to a user, especially in an environment where it is unsafe for a client to store cryptographic keys. Since Naor and Shamir introduced their first VCS in their seminal work in Naor and Shamir (1994), many researchers presented their own VCS's operating in various settings. Such latter schemes considered different parameters such as the number of visual shares, secret images, pixel expansion, computation constraints and image formats (Blundo et al., 2000; Hsu et al., 2004; Wu and Chang, 2005; Chang et al., 2005; Fang and Yu, 2006; Shyu et al., 2007; Fang, 2007; Feng et al., 2008; Ulutas et al., 2008; Chen et al., 2008; Weir and Yan, 2009; Tan, 2009; Verheul and van Tilborg, 1997; Yang and Laih, 2000; Chang et al., 2000; Chang and Yu, 2002; Lukac and Plataniotis, 2005; Shyu, 2006; Zhang et al., 2008; Wang et al., 2009; Lee and Le, 2009; Wei Qiao and Liang, 2009; Lee and Chiu, 2013). Among them, Naccache (1994) proposed an interesting visual cryptography scheme using colors instead of black and white patterns. Later, Rijmen and Preneel (1996) proposed a method (called RP hereafter) that took advantage of mixing colors as we did in the design of our scheme. Although both RP and our scheme share the idea of mixing colors to show another color, they are different in two aspects. First, RP was not designed to work on the usage model defined for our scheme, where one share is printed on a film and the other is on a light-emitting screen. To this end, while our system uses additive and subtractive color systems simultaneously, RP uses the additive color system only. Furthermore, RP is used for representing multiple colors, whereas our scheme is used only for representing the black color. Second, while RP divided a pixel into four subpixels to let the human visual system average out a color combination of multiple colors, our scheme does not use subpixels. As a result, our scheme is capable of showing information in full resolution on a fixed size screen, whereas RP has a resolution reduction of four due to the usage of subpixels. Following the Naccache and Rijmen papers, a thread of papers building upon them were published (Ateniese et al., 2001; Blundo et al., 1999; Naor and Pinkas, 1997; Blundo et al., 1998; Encinas et al., 2002; Nakajima and Yamaguchi, 2002; Jin et al., 2005; Koga et al., 2001; Tsai et al., 2007). However, none of these works describe an idea identical or similar to the one introduced in this work in terms of the two aspects described above.

While one may consider using any of those previous schemes as an off-the-shelf solution to the problem at hand, existing VCS's fall short because they are intended for one-time use, and therefore they do not allow the use of a share for multiple times. This missing feature is important for authentication protocols, particularly in Internet banking. Notice that a (k,n) scheme (Fang and Yu, 2006; Lee and Le, 2009; Ulutas et al., 2008; Verheul and van Tilborg, 1997; Lee and Chiu, 2013) does not mean the scheme can be used for multiple times. Other than this essential problem, the existing VCS's have the following issues.

First, some VCS's require computations for decryption to enhance their performance (Yang and Laih, 2000; Lukac and Plataniotis, 2005; Chang et al., 2000; Tsai et al., 2009; Heidarinejad et al., 2008). However, since using a hardware is not allowed in decryption phase in light of our system model, VCS's that require computations cannot be used as a solution to the problem at hand.

Second, in most VCS's, there is no restriction on shares' resolution, thus image quality is guaranteed by freely expanding pixels of the shares. However, the resolution of an image in our solution is restricted because of limited sizes of the VCC and VCC blocks. The size of the VCC is limited by portability and by compatibility with mobile banking on a smartphone with small screen. For compatibility reasons, we decided to use the size of a credit card. Also, the size of a block of the VCC needs to be reasonably big, to provide compatibility with different dot pitches of different screens. As a consequence of the sizes of the block and the VCC, pixel expansion is hard to use and only low resolution is allowed, which is not favorable to most VCS's. Even though there are several VCS's that do not use pixel expansion in the literature (Ito et al., 1999; Tzeng and Hu, 2002; Chen et al., 2007; Hou and Tu, 2005; Yang, 2004), those schemes already assume high resolution of shares.

7. Conclusion and future work

Internet banking is one of the most sensitive services in which security is a priority. In certain circumstances, such as operating on a compromised client, cryptographic protocols cannot guarantee security against the adversary who can control the client with a malware. As an example, we described and showed transaction-manipulation attacks and their applicability to this scenario.

To address this issue, we proposed the first transaction authentication protocol using a novel visual encryption scheme that utilizes complementary colors. We demonstrated usability and security features of our design using analytical and experimental results. Because the proposed scheme requires no computational aid but only low resolution for the visual cryptography card, it is usable for various settings including both a PC and a mobile computing environment. However, it should be noted that the population of participants in our user study is limited. Therefore, it is not clear whether the proposal might be applicable also to a person with very low visual acuity or elderly person. It would be an interesting future research topic to apply the proposal to a wide range of users and analyze the usability. If the proposed scheme is not applicable to some people, a possible

deployment scenario would be to allow a user to choose either a traditional authentication method or ours, which guarantees compatibility with the existing systems.

REFERENCES

Ateniese G, Blundo C, Santis AD, Stinson DR. Extended capabilities for visual cryptography. Theor Comput Sci 2001;260:143–61.

Blundo C, D'Arco P, Santis AD, Stinson DR. Contrast optimal threshold visual cryptography schemes. SIAM J Discret Math 1998;16:224–61.

Blundo C, Santis AD, Stinson DR. On the contrast in visual cryptography schemes. J Cryptol 1999;12:261–89.

Blundo C, Santis AD, Naor M. Visual cryptography for grey level images. Inf Process Lett 2000;75(6):255–9.

Chang C-C, Yu T-X. Sharing a secret gray image in multiple images. In: International Symposium on Cyber Worlds, 2002. Proceedings; 2002. p. 230–7.

Chang C-C, Tsai C-S, Chen T-S. A new scheme for sharing secret color images in computer network. In: International Conference on Parallel and Distributed Systems; 2000. p. 21–7.

Chang C-C, Chuang J-C, Lin P-Y. Sharing a secret two-tone image in two gray-level images. In: International Conference on Parallel and Distributed Systems, 2005. Proceedings, vol. 2; 2005. p. 300–4.

Chen YF, Chan Y-K, Huang C-C, Tsai M-H, Chu Y-P. A multiple-level visual secret-sharing scheme without image size expansion. Inf Sci 2007;177(21):4696–710.

Chen T-H, Tsao K-H, Wei K-C. Multiple-image encryption by rotating random grids. In: International Conference on Intelligent Systems Design and Applications, 2008. ISDA'08, vol. 3; 2008. p. 252–6.

Color wheel. (2014).http://en.wikipedia.org/wiki/Color_wheel.

Drimer S, Murdoch SJ, Anderson RJ. Optimised to fail: card readers for online banking. In: Financial Cryptography and Data Security, vol. 5628; 2009. p. 184–200.

Encinas LH, del Rey Ãngel MartÃn, Encinas AH. Encryption of images with 2-dimensional cellular automata. In: Proc. 6th Multiconference on Systemics; 2002. p. 471–6.

Entrust IdentityGuard. (2014) [Online], http://www.entrust.com/gridcard/.

Fang WP. Visual cryptography in reversible style. In: Intelligent Info Hiding and Multimedia Signal Proc, vol. 1; 2007. p. 519–24.

Fang L, Yu B. Research on pixel expansion of (2, *n*) visual threshold scheme. In: International Symposium on Pervasive computing and Applications; 2006. p. 856–60.

Feng J-B, Wu H-C, Tsai C-S, Chang Y-F, Chu Y-P. Visual secret sharing for multiple secrets. Pattern Recognit 2008;41(12):3572–81.

Heidarinejad M, Yazdi AA, Plataniotis KN. Algebraic visual cryptography scheme for color images. In: IEEE International Conference on Acoustics, Speech and Signal Processing, 2008. ICASSP 2008; 2008. p. 1761–4.

Hiltgen AP, Kramp T, Weigold T. Secure internet banking authentication. In: IEEE Security & Privacy, vol. 4; 2006. p. 21–9.

Hou Y-C, Tu S-F. A visual cryptographic technicque for chromatic images using multi-pixel encoding method. J Res Pract Inf Technol 2005;37(2):179–91.

Hsu H-C, Chen T-S, Lin Y-H. The ringed shadow image technology of visual cryptography by applying diverse rotating angles to hide the secret sharing. In: IEEE International Conference on Networking, Sensing and Control, vol. 2; 2004. p. 996–1001.

IBM Zone Trusted Information Channel (ZTIC). (2014) [Online], http://www.zurich.ibm.com/ztic/.

Ito R, Kuwakado H, Tanaka H. Image size invariant visual cryptography. IEICE Trans Fundam 1999;E82-A(10):2172–7.

Jin D, Yan W-Q, Kankanhalli MS. Progressive color visual cryptography. J Electron Imaging 2005;14:224–61.

Kasiski F.W., *Die Geheimschriften und die Dechiffrir-Kunst*. Berlin: E.S. Mittler und Sohn.

Koga H, Iwamoto M, Yamamoto H. An analytic construction of the visual secret sharing scheme for color images. IEICE Trans Fundam Electron Commun Comput Sci 2001;E84-A:262–72.

Lee J-S, Le THN. Hybrid (2,n) visual secret sharing scheme for color images. In: International Conference on Computing and Communication Technologies, 2009. RIVF'09; 2009. p. 1–8.

Lee K-H, Chiu P-L. Image size invariant visual cryptography for general access structures subject to display quality constraints. Image Process IEEE Trans 2013;22(10):3830–41.

Lukac R, Plataniotis KN. Bit-level based secret sharing for image encryption. Pattern Recognit 2005;38(5):767–72.

Naccache D. Colorful cryptography - a purely physical secret sharing scheme based on chromatic filters. In: French-Israeli Workshop on Coding and Information Integrity; 1994.

Nakajima M, Yamaguchi Y. Extended visual cryptography for natural images. J WSCG 2002;10:303–10.

Naor M, Pinkas B. Visual authentication and identification. In: Advances in Cryptology CRYPTO'97; 1997. p. 322–36.

Naor M, Shamir A. Visual cryptography. In: EUROCRYPT; 1994. p. 1–12.

Oppliger R, Rytz R, Holderegger T. Internet banking: client-side attacks and protection mechanisms. IEEE Comput 2009;42(6):27–33.

Rijmen V, Preneel B. Efficient colour visual encryption or shared colors of benetton. In: EUROCRYPT'96 Rump Session; 1996.

Shyu SJ. Efficient visual secret sharing scheme for color images. Pattern Recognit 2006;39(5):866–80.

Shyu SJ, Huang S-Y, Lee Y-K, Wang R-Z, Chen K. Sharing multiple secrets in visual cryptography. Pattern Recognit 2007;40(12):3633–51.

Starnberger G, Froihofer L, Göschka KM. Qr-tan: secure mobile transaction authentication. In: International Conference on Availability, Reliability and Security, 2009. ARES'09, vol. 5628; 2009. p. 578–83.

State-of-the-art technology – the best way to ensure secure e-banking. (2014) [Online], http://www.ubs.com/ch/en/online-services/security.html.

Tan X qing. Two kinds of ideal contrast visual cryptography schemes. In: 2009 International Conference on Signal Processing Systems; 2009. p. 450–3.

Tsai D-S, Chen T-H, Horng G. A cheating prevention scheme for binary visual cryptography with homogeneous secret images. Pattern Recognit 2007;40:2356–66.

Tsai D shiau, Horng G, Chen T, Huang Y-T. A novel secret image sharing scheme for true-color images with size constraint. Inf Sci 2009;179(19):3247–54.

Tzeng W-G, Hu C-M. A new approach for visual cryptography. Des Codes Cryptogr 2002;27(3):207–27.

Ulutas M, Yazc R, Nabiyev VV, Ulutas G. (2,2)-secret sharing with improved share randomness. In: International Symposium on Computer and Information Sciences, 2008. ISCIS'08; 2008. p. 1–5.

Verheul ER, van Tilborg HCA. Constructions and properties of k out of n visual secret sharing schemes. Des Codes Cryptogr 1997;11(2):179–96.

Wang D, Yi F, Li X. On general construction for extended visual cryptography schemes. Pattern Recognit 2009;42(11):3071–82.

Wei Qiao HY, Liang H. A kind of visual cryptography scheme for color images based on halftone technique. In: 2009. ICMTMA'09. International Conference on Measuring Technology and Mechatronics Automation; 2009. p. 393–5.

Weigold T, Kramp T, Hermann R, Höring F, Buhler P, Baentsch M. The zurich trusted information channel - an efficient defence against man-in-the-middle and malicious software attacks. In: International conference on Trusted Computing and Trust in Information Technologies: Trusted Computing – Challenges and Applications; 2008. p. 75–91.
Weir J, Yan W. Sharing multiple secrets using visual cryptography. In: IEEE International Symposium on Circuits and Systems, 2009. ISCAS 2009. IEEE; 2009. p. 509–12.
Wu H-C, Chang C-C. Sharing visual multi-secrets using circle shares. Comput Stand Interfaces 2005;28(1):123–35.
Yang C-N. New visual secret sharing schemes using probabilistic method. Pattern Recognit Lett 2004;25(4):481–94.
Yang C-N, Laih C-S. New colored visual secret sharing schemes. Des Codes Cryptogr 2000;20(3):325–36.
Zhang H, Wang X, Cao W, Huang Y. Visual cryptography for general access structure using pixel-block aware encoding. JCP 2008;3(12):68–75.

YoungJae Maeng received the BS and MS degrees in Computer Science and Information Technology from Inha University in 2006 and 2008, respectively. He is currently a Ph.D. candidate in Computer Science and Information Technology from Inha University. From 2012, he has been a member of engineering staff at The Attached Institute of Electronics and Telecommunications Research Institute, Korea. His research interests include network security, usable security, and their applications to authentication.

Aziz Mohaisen obtained his Ph.D. degree in Computer Science from the University of Minnesota, in 2012. He is currently a Senior Research Scientist at Verisign Labs. Previously, he was a Member of Engineering Staff at the Electronics and Telecommunication Research Institute, a large research and development institute in South Korea. His research interests are in the areas of networked systems, systems security, data privacy, and measurements. He is a member of IEEE and ACM.

Mun-Kyu Lee received his BS and MS degrees in Computer Engineering from Seoul National University in 1996 and 1998, respectively, and his PhD degree in Electrical Engineering and Computer Science from Seoul National University in 2003. From 2003 to 2005, he was a senior engineer at Electronics and Telecommunications Research Institute, Korea. He is currently an associate professor in the Department of Computer and Information Engineering at Inha University, Korea. His research interests include information security and theory of computation.

DaeHun Nyang received a B.Eng. degree in electronic engineering from Korea Advanced Institute of Science and Technology, M.S. and Ph.D. degrees in computer science from Yonsei University, Korea in 1994, 1996, and 2000 respectively. He has been a senior member of the engineering staff at Electronics and Telecommunications Research Institute, Korea, from 2000 to 2003. Since 2003, he has been a full professor at Computer Information Engineering Department of Inha University, Korea where he is also the founding director of the Information Security Research Laboratory. He is a member of the board of directors and editorial board of Korean Institute of Information Security and Cryptology. Dr. Nyang's research interests include cryptography and network security, privacy, usable security, biometrics and their applications to authentication and public key cryptography.