Display Integrity Assurance for SMS Transaction Authorization

Mohammed Alzomai
Queensland University of Technology

Audun Jøsang
University of Oslo, Norway

Bander Alfayyadh
Queensland University of Technology

Abstract

Secure online transactions with human users normally require visual display for verifying the correctness of central elements of the transaction before it is submitted. When commodity computer platforms get exposed to the Internet, even for a short period, there is a real and substantial risk that they become infected with malware that can modify anything on the computer, including what is displayed to the user and what is being sent over the Internet. This threat makes visual verification of transaction data unreliable and undermines other security mechanisms used to protect online transactions. This paper proposes a secure optical inspection technique for verifying the integrity of transaction data in online transactions. This technique provides protection against the highly advanced threat of a malware that manipulates transaction data as well as displayed data on the visual display of the client platform.

1. Introduction

Users generally rely on what they see on a computer display to read the output of transactions, to verify that they type correctly, and to ensure that the data being sent through online interface is according to their intentions. In general, all this depends on the integrity of the computing platform to which the VDU (Visual Display Unit) is connected. In practice it is extremely difficult to assess the integrity of a general purpose computing platform, and thereby to ensure that what the VDU displays is correct [2, 11, 12, 16, 17, 18].

The prospect that the computer display can lie to users is both frightening and real. This problem is amplified by the fact that people often read data from platforms that are not under their control, and that there are financial incentives for trying to manipulate the systems and the way data is displayed.

This paper proposes a method for display security to make the verification of displayed data in the SMS transaction authorization scheme more robust against the threat of compromised platforms. It is a method for assuring the correctness of displayed data in online transactions, i.e. to ensure that what is displayed on the VDU corresponds to what is being transmitted to other parties in online transactions. It assumes that the user has a PDA (Personal Digital Assistant) with an integrated camera, OCR (Optical Character Recognition) and communication functions. The method is based on using a portable PDA/camera (e.g. mobile phone) to capture the data from the VDU, recovering the data from the image through OCR, and using an out-of-band channel for matching this data with the data received by the transaction partner. In order to successfully falsify data by attacking the platform integrity, the attacker needs to have compromised both the client platform and the PDA, which is more difficult than to only compromise one of them. The proposal therefore provides a robust method for verifying displayed data because it is considered harder to compromise both platforms.

2. Related Work

A system with a display security assurance concept developed by Cronto Limited [13] already exist in the marketplace. Cronto system is based on displaying encrypted data rather than clear text data.

The Cronto system, which is based on research undertaken at the University of Cambridge, provides a transaction authentication solution for online banking that takes advantage of the camera in customers' mobile phones. The Cronto solution is based on capturing a visual cryptogram sent by the online bank server and extracting the transaction details from it [6].

The scheme starts when the customer initiates a new transaction with the online bank. Upon receiving the transaction request, a Cronto server-based software module in the online bank side takes the requested transaction details from the banking application and generates a unique visual cryp-

togram challenge which hides the transaction details and then passes the cryptogram to the customer for authentication. The server then validates the client's response and determines whether the transaction should be authorized or not.

At the user side, the customer uses the camera of his/her mobile equipped with the Cronto client software to capture the visual cryptogram provided by the bank, extract the transaction details from it, verify the transaction details and confirm the transaction by entering a generated code into the client browser.

2.1 Cronto System Architecture

The system comprises an access terminal which can be a network connected computer, a remote authentication device such as a network server and a mobile device such as a mobile telephone or a Personal Digital Assistant (PDA).

As shown in Figure 1, the user starts the authentication process in order to initiate a specific transaction by entering his/her personal credentials such as a user name and a password into the access terminal (step 1). Next, the user credentials are sent by the access terminal to the remote authentication device through Internet channel (step 2) along with transaction details which the user wishes to perform.

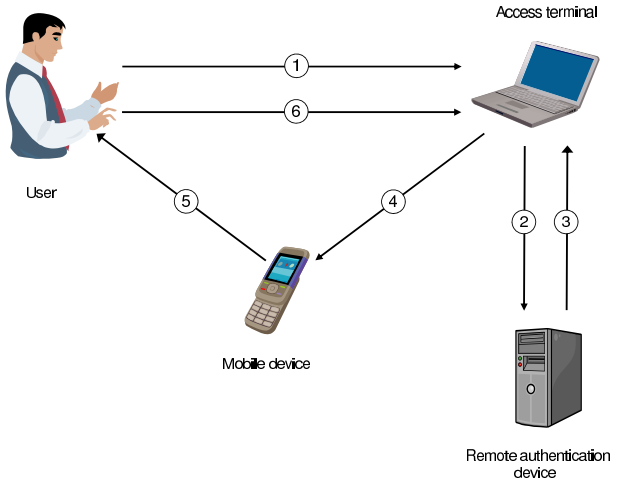

Figure 1. Cronto security system scenario

The remote authentication device then validates the user credentials. If the user credentials are successfully validated, the remote authentication device then generates a message which is encrypted. The message may be partially based on a randomly generated code, such as a numeric code as well as transactional details. Adding transactional details provides a transaction integrity measure which aims at protecting against attacks such as a "man-in-the middle" where attackers are prevented from modifying the content of the transaction without being detected. This is in contrast to a system relying only on dynamic passwords (e.g. One-Time Passwords) which do not carry any form of transaction details and are independent of the transaction.

The encrypted message is embedded into a cover image signal so that the original signal and modified signal are perceptually indistinguishable. The encrypted signal can be embedded into a cover image signal using any known form of steganography[1] or digital watermarking [6].

The modified signal is then sent via Internet to the access terminal (step 3) and is displayed on its screen. The user then uses the camera on his/her mobile device to capture the image from the access terminal (step 4). An application installed on the mobile device will process the image in order to extract the encrypted message. The extracted message is then decrypted and the generated code and transaction details are then displayed to the user (step 5). If the user verifies the transaction details and is satisfied with the correctness of the transaction, he/she may confirm the transaction by entering the code into the access terminal (step 6). The code is then sent to the remote authentication device for the purpose of comparing it to the code which was originally generated by the remote authentication device. If the two codes match, the transaction is successfully authenticated.

3. Prior Art: SMS Transaction Authorization

A method often used for verifying the correctness of online bank transaction data consists of sending the data with an authorization code by SMS to the user's mobile phone. This enables the user to verify that the transaction data assumed by the bank are according to the user's intentions, and to confirm the transaction. For this to work the user must manually copy the authorization code from the mobile phone display to the client platform and submit it to the online bank server as a confirmation of the transaction. The scenario is illustrated in Figure 2, where the numbered circles indicate the sequential order of the actions/messages described in Table 1.

The main advantage of the SMS transaction authorization is that SMS messages sent from the bank to the user's mobile phone pass through the cellular network, which is separate and independent from the Internet. By verifying the authorization code received from the client platform, the bank can conclude that the user received the SMS message through the cellular network, read it and submitted it through the Internet. This is then interpreted as a genuine intent to confirm the transaction. The security of this scheme is based on the assumption that it is difficult for an attacker to steal the user's personal mobile phone and to attack the cellular network [10].

Assuming a so-called man-in-the browser attack, i.e. that an attacker changes the amount and/or the destination account

[1] Steganography is a form of security through obscurity science where a hidden message is written in a mode that no one other than the sender and recipient suspects its existence. The message may be concealed in forms such as images or text.

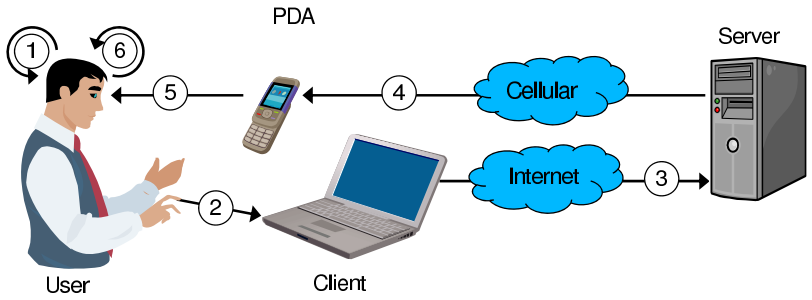

Figure 2. SMS-based transaction integrity check

♯	Message/Action description
1.	The user types and inspects the transaction data displayed on the client VDU
2.	The user initiates the transaction from the client platform
3.	The transaction data are sent by the client to the server
4.	The assumed transaction data with authorization code are sent as SMS from the server to the user's mobile
5.	The assumed transaction data and authorization code from the SMS are displayed
6.	The user reads the assumed transaction data which enables him to make a conclusion about the integrity of the transaction. In the positive case the user copies and submits the authorization code to confirm the transaction. In the negative case the user aborts the transaction.

Table 1. Messages in the SMS-based transaction integrity check scenario

number by a Trojan program on the client platform, the modified amount and account number will appear in the SMS message. The scheme relies on user awareness, and it is assumed that the correctness of the amount and of the destination account number is verified by the user before copying the authorization code from the SMS message. Assuming that the user verifies the correctness of the amount and of the bank account number in the SMS message, this scheme is secure against attacks on the client platform, and is in fact independent of the security of the client platform. This represents a considerable security improvement. However, if a user victim fails to notice that the bank account number in the SMS message is not the same as the intended account number, the attack will succeed.

While the mental load of verifying the correct amount and destination account specified by the SMS message is probably acceptable for a single transaction, the repeated process of verifying the same for each transaction can be quite tedious and therefore lead to user apathy. It has been noted that when faced with a frustrating security task, users may usually bypass or ignore that task [1, 5, 15].

In an experiment [3] we studied the usability of the SMS transaction authorization scheme by observing whether users are able to perform the extra tasks of verifying the correctness of transaction detail. This is important because banks would normally assume that users are responsible for transactions authenticated with the authorization code. However, if a significant proportion of users are unable to use the method correctly, this assumption would be unreasonable and should be reassessed by the banks.

According to the study [3] about 21% of realistic attacks were successful, meaning that 21% of the users failed to notice that half the digits in the destination account number had changed. This in our opinion represents an inadequate level of security for the SMS transaction authorization system. The study also found that a *stealthy attack*, where only one out of eight digits of the destination account number is altered, was successful in 61% of the attacked transactions. This shows that as the number of altered digits decreases the success rate of attacks increases. In general this reflects a fundamental limitation in the user's ability to reliably verify long strings of data. The validation process will even be more difficult with the trends to use the International Bank Account Number (IBAN). The IBAN is an international standard for identifying bank accounts aiming at minimizing the risk of propagating transcription errors and can consist of up to 30 digits.

To enhance the SMS transaction authorization scheme without compromising the strong authentication process of validating every transaction, the validation process has to be automated. The burden caused by the extra task of validating

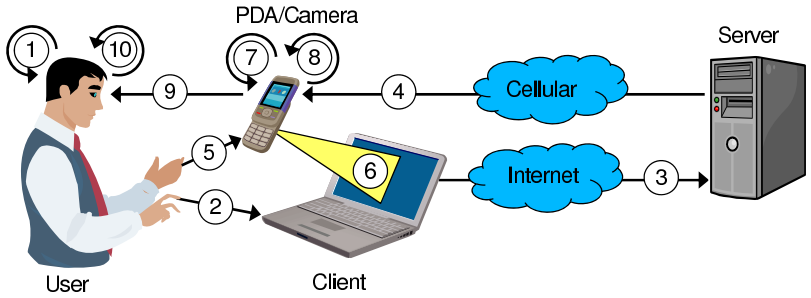

Figure 3. Display security architecture scenario

♯	Message/Action description
1.	The user types and inspects the transaction data displayed on the client VDU
2.	The user initiates the transaction from the client platform
3.	The transaction data are sent by the client to the server
4.	The assumed transaction data with authorization code are sent as SMS from the server to the user's mobile phone
5.	The user activates the camera function on the PDA
6.	A photo is taken of the text displayed on the VDU
7.	The OCR function in the PDA recovers the text from the photo
8.	The PDA compares the data from photo and from server
9.	The PDA signals the success/failure of the comparison
10.	The user receives the signal from the PDA and can conclude about the integrity of the displayed/transmitted transaction data

Table 2. Messages and actions in the display security architecture

every transaction in a sufficient manner can be shifted to the mobile phone. The user will never need to manually revalidate the correctness of the transaction details after they have been typed on the client terminal; instead the verification process can be executed by the mobile phone. The next section describes the system outline of the proposed scheme.

4. The Display Security Architecture

The main idea of the new enhanced scheme is to use a personal portable platform that is able to convert a picture of the text-based representation of transaction taken from the VDU into its character-based representation using OCR (Optical Character Recognition) software. The conversion from picture to characters first requires the optical image emitted from the VDU to be captured by a digital camera. The bitmap representation of the image produced by the digital camera is then translated to a digital data representation by an OCR process.

To be more precise, the idea is to check the transaction data received through e.g. an SMS message against the trusted transaction data extracted from the visual representation of a transaction data on the VDU. So, when the user starts a transaction involving a bank account and receives a confirmation SMS from the bank, a validation application on the PDA will be activated to validate the transaction details. The validation application will compare the received transaction data in the SMS against the data extracted from the photo. As a result of the comparison, the PDA will either display a message confirming validity or display a message warning the user about an invalid transaction data.

4.1 Scenario

The proposed scenario is a modified version of the SMS transaction authorization scheme illustrated in Figure 2 and Table 1 in Section 3. In the new scenario, steps 5 and 6 were substituted with the new steps 5, 6, 7, 8, 9, and 10 to allow the auto validation process of the transaction data. In the new steps, the PDA checks the transaction data received in the SMS against the transaction data extracted from the analogue visual representation of the transaction data in the display of the client platform. Also, the PDA will signal the success or failure of the comparison to the user who can then make a conclusion about the integrity of the transaction data.

As illustrated in Figure 3 where the numbered circles indicate the sequence of messages and actions, the scenario is

initiated when the user starts a new transaction and enters the transaction details and at the same time inspects the entered data displayed on the client's VDU (steps 1 and 2). The transaction request and the transaction data are then sent to the bank server (step 3). Upon receiving the transaction request, the bank server will send the assumed transaction details with an authorization code to the user. The data will be sent as an SMS message to the user's mobile phone through the cellular network (step 4).

After receiving the SMS from the bank, the user will activate the mobile phone's camera and capture the transaction details displayed on the client VDU (steps 5 and 6). Next, an OCR function on the user's mobile phone will convert the image of the transaction data into a readable character format (step 7). The mobile phone will then compare the transaction details extracted from the image with the transaction data in the received SMS message (step 8).

Finally, the mobile phone will either signal a successful match and present the authorization code to the user or display a warning message about a failure comparison (step 9). At this point the user can make a conclusion about the security and integrity of the displayed and transmitted data (step 10). The scenario is illustrated in Figure 3 and summarized in Table 2.

4.2 Practical Aspects

Mobile phones commonly have integrated digital cameras, so they already have the necessary functional basis for the proposed method. The inclusion of software for the OCR and for comparing transaction data is all that is needed.

Commercial and open source OCR software packages are available. In its simplest form, OCR software takes scanned documents and converts them into text files. More advanced graphical layout of digital documents will require a standard for geometrically formatting documents so that the translation from graphical bitmap format to digital character format is unambiguous.

With current technology, many mobile phones are equipped with OCR capabilities. An example of using one of the existing mobile phone's OCR functions to convert the data displayed on the client's VDU to an editable text format is illustrated below.

Example: Image of transaction data to transaction text data

Figure 4 shows a screen shot of a sample transaction taken from a simulated Web page of a real bank (The commonwealth bank, Australia). It displays the transaction details of a funds transfer to be transmitted to the online bank.

Figure 5 shows a snapshot photo of a simulated portion of the bank Web page that shows the transaction details to be verified. The photo is taken by an iPhone 3GS mobile with OS version 3.1.2 and is equipped with an auto focus 3 mega pixels camera [7].

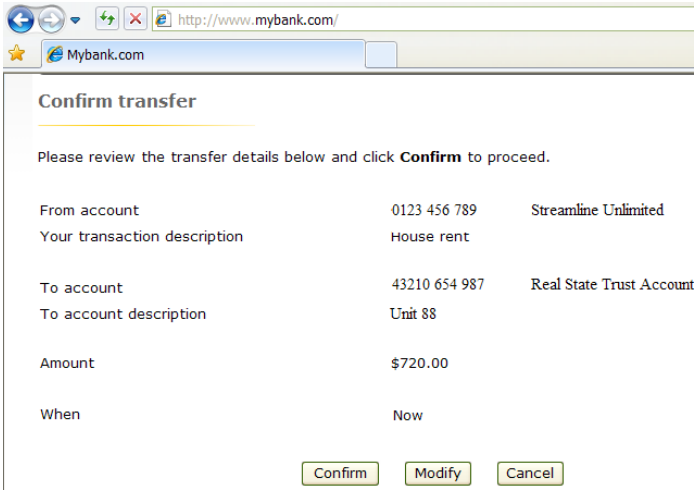

Figure 4. A screen shot of a transaction data

The transaction details were shown in a browser default font size of the web page with black font color and white background. The display unit was a 22" Dell Widescreen LCD Monitor model no. 2208WFPt. The resolution was 1280 by 768 pixels. The display brightness and contrast setting were %60. The person who took the photo was setting on a chair in front of the display and was holding the mobile phone camera in a straight angle to the centre of the display. The distance between the display and the mobile phone camera was about 20cm.

Figure 5. A photo of a transaction data

Figure 6 shows the transaction text data resulting from applying an OCR function on the photo of Figure 5 executed by one of the iPhone's OCR applications called *ocrNow!* that converts images photographed on the iPhone camera into text

[9]. Many other mobile applications with built in OCR capabilities are also available.

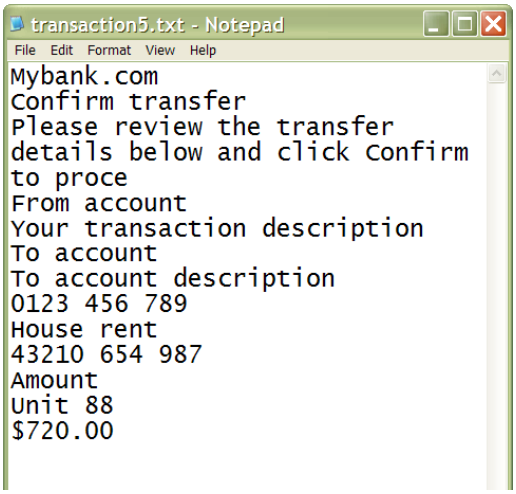

Figure 6. Text data resulting from converting a transaction image

Because taking a good photo of the visual display could easily become a non-trivial task, the integrity of the proposed solution was examined. Participants were invited to carry out the tasks of the practical example presented above. There were 24 participants who are categorized into four groups. The first group consists of seven participants aged 20 to 30 years. The second group includes six participants aged 31 to 40 years. The third group contains six participants as well but they were aged 41 to 50 years. The last group comprises five participants aged 51 years and older; see Table 3.

Group	Age	participants
G1	20-30	7
G2	31-40	6
G3	41-50	6
G4	51+	5
Total		24

Table 3. Participants groups

The experiment was conducted in two stages. In the first stage, each participant used an iPhone 3GS mobile phone camera and undertook five trials of the above example. As a result, the total number of trials was 120 and the success rate of converting the transaction data from the graphical format to a valid text format was around 81% (97 out of 120). See Table 4.

In the second stage, each participant was asked to repeat the trial five times using an iPhone 4 auto focus 5-megapixel still image camera [8]. The second stage experiment resulted also in 120 trials where the success rate of converting the transaction data from the graphical format to a valid text format was around 86% (103 out of 120). See Table 5.

Group	Photos	Success	%	Failure	%
G1	35	30	86	5	14
G2	30	26	87	4	13
G3	30	23	77	7	23
G4	25	18	72	7	28
Total	120	97		23	

Table 4. Results using iPhone 3GS camera

Group	Photos	Success	%	Failure	%
G1	35	32	91	3	9
G2	30	27	90	3	10
G3	30	25	83	5	17
G4	25	19	76	6	24
Total	120	103		17	

Table 5. Results using iPhone 4 camera

4.3 Security Analysis

The security of the proposed system requires that the client platform and the PDA are not both compromised concurrently. Assuming that the client platform and the PDA have not both been compromised concurrently, it is possible to verify that the PDA indeed provides the necessary elements for a robust transaction data verification process.

Assuming that the client platform has been compromised; for example, an interjected Man-in-the-Browser malware can alter the data between the user and the web browser's security mechanism in a completely surreptitious way. When the user sends a transaction request to the online bank, the malware can modify the transaction details and pass the altered transaction to the online bank without the user's knowledge. The online bank will receive the request with the altered transaction and cannot differentiate between the original transaction and the fake one. The online bank then sends the fake transaction details as an SMS to the user's mobile. In this case the wrong transaction data is sent in message (3) and consequentially in message (4). However, the comparison between the received transaction data in the SMS and the digital data converted from the image by the PDA in step (8) will fail, so that the PDA will signal that the transaction data have been altered.

If the PDA is compromised alone, this will not affect the system security since the client platform will transmit the correct transaction data. However, it is possible that the PDA generates a false alarm.

Considering now the possible consequence of a double compromise, i.e. that the client platform has been compromised so that it sends the wrong transaction data to the server in (3), and that the PDA has been compromised so that it

wrongfully validates the transaction data in step (9); in this case, the user will instruct the client platform to confirm the altered transaction, so that the attack will succeed.

It thus requires compromise of both the client platform and the PDA in order to break the security of the proposed system.

The strength of the proposed system is based on minimizing attack possibilities and managing the risk. Attacks are reduced by separating the transaction validation process (executed in the PDA) from the transaction creation and execution process which is performed in the client platform.

The advantage of the proposed method is that the visual comparison of transaction data intended by the user and assumed by the server is automated, and that this validation process is separated from the client platform where the transaction is managed. Commodity client platforms are typically designed with priority on flexibility and functionality, which unavoidably results in security vulnerabilities.

The security property is based on using a digital camera which "sees" the transaction data to be transmitted exactly as the user sees it. The bitmap image is then converted to the character format using OCR techniques. This bridges the semantic distance between the digital data in its binary form and the analogue visualisation of the data that the user perceives. It basically guarantees that what you see is what you intend to transfer.

4.4 Usability Analysis

The usability of the proposed system is tested against a set of security usability principles defined in [4]. These principles describe user interaction with security systems in terms of usability.

The security usability principles are divided into principles for security action and security conclusion which can be described as follows:

- A *security action* is when users are required to produce information and security tokens, or to trigger some security relevant mechanism. For example, typing and submitting a password is a security action.
- A *security conclusion* is when users observe and assess some security relevant evidence in order to derive the security state of systems.

The eight security usability principles are:

1. Security Action Usability Principles
   (a) The users must understand which security actions are required of them.
   (b) The users must have sufficient knowledge and the practical ability to make the correct security action.
   (c) The mental and physical load of a security action must be tolerable.
   (d) The mental and physical load of making repeated security actions for any practical number of transactions must be tolerable.
2. Security Conclusion Usability Principles
   (a) The user must understand the security conclusion that is required for making an informed decision. This means that users must understand what is required of them to support a secure transaction.
   (b) The system must provide the user with sufficient information for deriving the security conclusion. This means that it must be logically possible to derive the security conclusion from the information provided.
   (c) The mental load of deriving the security conclusion must be tolerable.
   (d) The mental load of deriving security conclusions for any practical number of service access instances must be tolerable.

In the SMS transaction authorization scheme described in Section 3 the mental load of repeatedly verifying several account numbers may violate principle 1d and represent a usability concern. As the experiment conducted in [3] showed, users were vulnerable to attacks due to this usability problem.

In the proposed system, usability is improved by delegating the account-number verification task to the PDA. The user only has to interpret the result of the comparison between the OCR generated file and the SMS received from the server. Clearly, this is an easier task to perform than comparing lengthy alpha-numeric numbers.

If the user performs repeated transactions which will require taking many snapshots of the VDU, this may become a usability issue but to a lesser extent than the issue associated with the SMS transaction authorization scheme. Taking several snapshots is less of a mental load than several comparisons of lengthy alpha-numeric numbers.

Usability can be enhanced by minimizing the number of transactions the user needs to validate. As an example, a combined system of our proposed solution and the scheme proposed in [4] can enhance the usability. The scheme in [4] works by maintaining several lists of accounts in the PDA generated from previous transactions. Examples of these lists are "Trusted Accounts list", "Malicious Accounts list", etc. The PDA would look into these lists first for comparison with the server's SMS before looking for a new OCR file. As the user continues to use the system, this can significantly reduce the number of snapshots a user needs to take and will not undermine the concept of verifying transaction data on an individual basis as every transaction data is checked by the PDA

either against the trusted accounts list (which is separate and independent from the client terminal) or by a new OCR. The combined system of the two proposed solutions is illustrated below:

1. The user types and inspects the transaction data displayed on the client VDU.
2. The user starts the transaction in the client platform.
3. The transaction data are sent by the client to the server.
4. The assumed transaction data with an authorization code are sent as an SMS to the user's mobile phone.
5. The mobile phone checks the account number in the SMS against the account lists.

   If the account number is found in the trusted list, the mobile phone:

   - Signals a message of a trusted account, and
   - Shows the transaction details and the authorization code.

   If the account is not in the trusted list, it will be searched in the malicious accounts list; if the account is found, the mobile phone:

   - Signals a warning message, and
   - Shows the transaction data without the authorization code.

   If the account number is not in the two lists:

   - The user activates the camera function on the mobile phone.
   - A photo of the text displayed on the VDU is taken.
   - The OCR function in the mobile phone recovers the ASCII text from the photo.
   - The mobile phone compares the data from the photo and from the server.
   - The mobile phone signals the success/failure of the comparison and shows the transaction data and the authorization code.
6. The user receives one of the above signals from the mobile phone and can draw a conclusion about the integrity of the displayed/transmitted transaction data.
7. The user either confirms or cancels the transaction and if asked, may add the account number to either the trusted or malicious list.

5. Discussion

The practicality of the proposed solution depends extensively on the capability of the mobile phone scale camera to take good images of the transaction data displayed in the client VDU as well as the integrity of the OCR system that converts these images into text format.

The successful rate of converting the transaction data from the graphical format to a valid text format was around 86% when using iPhone 4 mobile phone camera. This indicates that about one out of seven attempts to take a snapshot of the transaction details displayed in the client VDU will result in an image that will not convert to a valid transaction data text. However, with advancing technology, it is expected that the successful conversion rate will continue to increase. As an example, the successful conversion rate increased from 81% to 86% after switching from iPhne 3GS to iPhone 4 camera which can take more quality photos.

In addition to the camera capabilities, the photo quality is also affected by other factors related to the photo source and the person who takes the photo. The photo source is the transaction data displayed on the client's VDU. Factors such as the VDU's brightness, screen background and font size of the transaction data can impact the photo quality. The factors related to the user who takes the photo may include the manner in which the user holds the camera and the angle and distance between the camera and the VDU. The experiment described in Section 4.2 gave indication that the age of the user who takes the photo may also have an effect on the photo quality.

The Cronto system described in Section 2 requires that both parties involved in the authentication process (i.e. the costumer and bank server) be equipped with proprietary cryptographic tools in order for the system to work. In contrast to the Cronto system, the proposed solution enhances the SMS transaction authorization without the use of additional cryptography and can be applied directly at the costumer side without making any changes at the server side.

The security of Cronto system relies on the client platform not being compromised. If an attacker accesses the encrypted transaction data in the client platform, Cronto system will fail. On the other hand, the security of the proposed method only depends on either the client platform or the PDA being secure. In fact, one of them can be compromised without causing a risk of tricking the user into confirming online transactions to the fake accounts, so the security of the proposed method is independent of the security of the client platform. The PDA can be designed with priority on security, and with limited functionality and flexibility. The PDA is controlled by the user, so he does not have to rely on systems outside his control when conducting financial transactions. This feature will allow mobility where users can apply the new method to any system anywhere as long as the sys-

tem is able to connect to the online bank and execute financial transactions.

From a usability perspective, the user of Cronto system will still need to manually verify the transaction details in the user device and the client platform. The verification may involve a comparison of long strings of characters. The study in [3] shows that as the number of altered digits in the attacked account number decreases the success rate of attacks increases. This reflects a limitation in the user's ability to reliably verify long strings of characters. Using the proposed solution however, the length of the account number has no effect on the verification process; no matter how long or short the account number is, the mobile phone verifies it automatically.

The separation of the functionality in the PDA from that on client platform gives the SMS transaction authorization scheme its security strength but with current technology it would also be possible for attackers to control the PDA and gain access to its data. In fact, a perfectly secure system will never exist and there will always be weaknesses. For example, attackers can get access to the PDA if it is connected to the Internet or if its Bluetooth is enabled i.e. making it available for a connection. The relatively new attack known as *blue snarfing*, for example, allows intruders to gain access to Bluetooth enabled phones by exploiting a security flaw in the wireless protocol [14].

6. Conclusion

People usually rely on what they see on a computer display to ensure that the data being sent through online transactions is consistent with their intentions. Generally, this depends on the integrity of the computing platform to which the VDU (Visual Display Unit) is connected. Practically, ensuring platform integrity is very difficult. To date, it is almost impossible to validate the integrity of a general purpose computing platform, and thereby to ensure that what the VDU displays is correct. The validation task is difficult because people regularly read data from platforms that are not under their control and because of the complexity of general purpose computing platforms.

Secure online transactions with human users require visual display for verifying the correctness of the transaction details before submission. When executing online transactions, there is a real risk that computer platforms become infected with malicious software that can alter what is displayed to the user and what is being sent online. This threat makes visual verification of transaction data unreliable and weakens other security mechanisms used to protect online transactions.

In this paper a new solution aimed at improving the overall security of online transactions by providing display security has been proposed. This removes the cognitive burden of e.g. manually validating every transaction by visual comparison of transaction data in SMS transaction authorization schemes.

Current display security technologies are not able to provide high assurance of the integrity of displayed data. This paper has shown that it is possible to make the security of displayed data independent of the security of the client platform. This is achieved by using a mobile phone equipped with a camera in parallel with the client platform when executing online transactions. The display security is based on using a digital camera which "sees" the displayed transaction data exactly as the user sees it. The bitmap image of the transaction data is then converted back to the original digital form using OCR techniques, so that the displayed transaction data can be automatically compared with those received through SMS from the bank.

The security of the proposed solution is gained from the fact that it would require compromise of both the client platform and the PDA in order to break the security of the proposed system. The solution's strength is based on minimizing attack possibilities by separating the transaction validation process, which is executed in the PDA, from the transaction execution process which is performed in the client platform.

References

[1] A. Adams and M. A. Sasse. Users are not the enemy. *Communications of the ACM*, vol.42(no.12):40–46, 1999.

[2] A. Alsaid and C. Mitchell. Dynamic Content attacks on Digital Signatures. *Information Management & Computer Security*, 13(4):328–336, 2005.

[3] M. Alzomai, B. Alfayyadh, A. Josang, and A. McCullagh. An experimental investigation of the usability of transaction authorization in online bank security systems. In Ljiljana Brankovic and Mirka Miller, editors, *Sixth Australasian Information Security Conference (AISC 2008)*, volume 81 of *CRPIT*, pages 65–73, Wollongong, Australia, 2008. ACS.

[4] Mohammed Alzomai, Audun Josang, Adrian McCullagh, and Ernest Foo. Strengthening sms-based authentication through usability. In *ISPA '08: Proceedings of the 2008 IEEE International Symposium on Parallel and Distributed Processing with Applications*, pages 683–688, Washington, DC, USA, 2008. IEEE Computer Society.

[5] D. Balfanz, G. Durfee, D. K. Smetters, and R. E. Grinter. In search of usable security: five lessons from the field. *Security and Privacy Magazine, IEEE*, vol.2(no.5):19–24, 2004.

[6] I. Drokov, E. Punskaya, and E. Tahar. System and Method For Dynamic Multifactor Authentication. US Patent No. US 2008/0307515 A1.

http://www.freepatentsonline.com/y2008/0307515.html, December 2008.

[7] Apple Inc. iPhone-3gs specs. http://www.apple.com/iphone/iphone-3gs/specs.html, August, 2010.

[8] Apple Inc. iPhone4 specs. http://www.apple.com/iphone/specs.html, August, 2010.

[9] Wordcraft international limited. *ocrNow!*, an OCR iPhone's application. http://www.wordcraft.com, August, 2010.

[10] A. Jøsang, M. Alzomai, and S. Suriadi. Usability and Privacy in Identity Management Architectures. In *The Proceedings of the Australasian Information Security Workshop*, 2007.

[11] A. Jøsang, D. Povey, and A. Ho. What You See is Not Always What You Sign. In *Proceedings of the Australian UNIX and Open Systems Users Group Conference (AUUG2002)*, Melbourne, September 2002.

[12] K. Kain, S.W. Smith, and R. Asokanm. Digital Signatures and Electronic Documents: A Cautionary Tale. In *Proceedings of IFIP Conference on Communications and Multimedia Security*.

[13] Cronto Limited. The Cronto system. http://www.cronto.com, August, 2010.

[14] S. Pradhan, E. Lawrence, and A. Zmijewska. Bluetooth as an enabling technology in mobile transactions. In *ITCC '05: Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'05) - Volume II*, pages 53–58, Washington, DC, USA, 2005. IEEE Computer Society.

[15] M.A. Sasse. Computer Security: Anatomy of a Usability Disaster, and a Plan for Recovery. In *Proceedings of the Conference on Human Factors in Computing Systems (CHI2003), (Workshop on Human-Computer Interaction and Security Systems)*, 2003.

[16] K. Scheibelhoferm. Signing XML Documents and the Concept of What You See Is What You Sign. Masters thesis, Graz University of Technology, Austria, 2001.

[17] Adrian Spalka, Armin B. Cremers, and Hanno Langweg. The fairy tale of 'What You See Is What You Sign - Trojan Horse Attacks on Software for Digital Signatures. In *IFIP Working Conference on Security and Control of IT in Society-II (SCITS-II)*, Bratislava, Slovakia, June 2001.

[18] Arnd Weber. See What You Sign: Secure Implementations of Digital Signatures. In *Proceedings of the International Conference on Intelligence and Services in Networks*, pages 509–520, 1998.