DIGITALSEAL: A TRANSACTION AUTHENTICATION TOOL FOR ONLINE AND OFFLINE TRANSACTIONS

Changhun Jung[◇], Jeonil Kang[◇], Aziz Mohaisen[⋆], and DaeHun Nyang[◇]

[◇]Inha University [⋆]University of Central Florida

ABSTRACT

We introduce DigitalSeal, a transaction authentication tool that works in both online and offline use scenarios. DigitalSeal is a digital scanner that reads transaction information sent by an issuing entity of the DigitalSeal reader for authentication, and the information is encoded using a specially crafted bar-code. DigitalSeal views various pieces of transaction information for users to verify and proceed with transaction authentication. DigitalSeal is generic, and is capable of reading information viewed on paper, computer monitors (similarly, kiosk monitors), and mobile phones. A prototype of DigitalSeal is built using a Arduino UNO, four LLS05-A sensors, four TCRT5000 sensors, a 1602 LCD and a 9V battery.

***Index Terms*—** Authentication, offline transactions.

1. INTRODUCTION

One-time passwords (OTPs) are passwords valid for only one transaction or login session, and there has been a large body of work in the literature on designing OTPs. TOTP [14] creates the password using the time-synchronization, HOTP [13] crates the password using the data and HMAC, and OTP based on response-challenge [2] creates a password by sending and receiving the random number in between user and server. These OTPs are realized in multiple products, which come in both software and hardware versions. While widely used and are indeed usable, the majority of those approaches suffer from Man-in-the-Middle (MitM; especially when implemented in hardware) [16], and Main-in-the-Browser (MitB; especially when implemented in software).

Optical readers with Transaction Authentication Number (TAN) [10] are used to address the aforementioned problem. In particular, the Ezio optical reader [6, 3] is introduced, in which a user inserts his smart card into the reader to scan a flickering bar-code with transaction information on a monitor. Upon reading the bar-code and passing the information to the user, he creates a TAN. If valid, upon verification by the user's side, the user may use TAN for authenticating a transaction while addressing MitM, MitB and abort otherwise. While it addresses the issue at hand, optical readers like Ezio only work for online transactions, but not offline transactions.

To address the shortcomings of optical readers, we introduce DigitalSeal, a transaction authentication tool that works in both online and offline scenarios. DigitalSeal is a digital scanner that reads transaction information sent by an issuing entity of DigitalSeal reader for authentication, and encoded using a specially crafted bar-code. DigitalSeal views various pieces of transaction information for users to verify for authentication. Unlike Ezio, which requires a special smart card and reader, DigitalSeal does not require the flickering bar-codes and smart card, and is capable of reading information viewed on paper, computer monitors (similarly, kiosk monitors), and mobile phones. DigitalSeal is built using several off-the-shelf components, provides operationally reasonable accuracy of authentication and usability, even using an initial prototype. Also DigitalSeal works for offline transactions.

2. RELATED WORK

There are many OTP products in the market, both as software and hardware, using either time-based and challenge-response based techniques. The most popular type of OTP products is based on time-synchronization [14] between the authentication server and the client providing the onetime password, often valid only for 30-60 seconds. Google authenticator and FreeOTP fall into this category by implementing TOTP (using the time-based one-time password defined in RFC 6238). RSA's secureID [8] uses its propriety algorithm, which is basically a TOTP, which comes in both hardware and software OTP. Another type uses a challenge-response protocol [2] instead of a timer, where a random challenge value is displayed for authentication and a user should type in the value in its OTP device to get a valid response value. Dell's defender software implements both types [5]. While usable, all of these products are vulnerable to Man-in-the-Middle (MitM) attack, with hardware-based solutions, and Man-in-the-Browser (MitB), as with software-based solutions. This is because OTPs are inherently independent of the transaction information and thus it is possible for an attacker to use the intercepted OTP for other transactions within a valid time epoch.

Other authentication techniques, utilizing non-PIN based approaches, such as complementary colors [20], permutations [17], and visual representation [21], and addressing

advanced adversarial models [15] have been also explored.

MitM [18, 15] is an attack technique that eavesdrops or manipulates the communication between two users. The attacker breaks in between the two users connecting the communication, and normally delivers the messages of the two users to each other. So two users think that connected to the each other, but actually both users are connected to the attacker. Attacker can attack by transmitting information to one side after eavesdropping and manipulating information transmitted from one side.

The MitB attack [4] is a technique of attacking a web page by using a web browser security vulnerability. This can cause harm to the user by by modifying transaction infomation or inserting additional transactions. MITB attacks are among the major threats to online banking systems in particular.

To address this problem, the most relevant approach is Gemalto's security tool called Ezio optical reader [6] which generates TAN [10]. A user inserts his own smartcard (credit or debit card) into the security tool to scan a flickering barcode sending a transaction information on a monitor. After receiving all the transaction information, the optical reader forwards it to the smartcard to generate a TAN [3]. If TAN is made correctly, the user can use it to authenticate on any transaction. By using the transaction information for generating TAN, Gemalto's Ezio can successfully defeat the MitM, MitB attacker. The limitation of Gemalto's approach, however, is that it can be used only for online transactions because it reads optical signal by flickering barcodes, while DigitalSeal does not require the smartcard, the flickering barcodes, and works also for offline transactions.

3. DIGITALSEAL

DigitalSeal is an authentication hardware tool for online and offline transactions. In DigitalSeal, a user swipes DigitalSeal downward so that it can scan a barcode on a screen or on a piece of paper. The barcode contains the transaction information, such as recipient name, amount of transaction, etc., and displays the HOTP's [11, 12] tag calculated with a preshared key and transaction data on DigitalSeal's LCD screen. For user's convenience, the barcode automatically scrolls on a screen of a smartphone and a computer so that a user can just place DigitalSeal on a screen instead of swiping.

3.1. Bar-code for DigitalSeal

We designed a dedicated bar-code system for DigitalSeal. The bar-code consists of square cells, and each cell is used to represent one bit. A black cell represents 1, while a white cell 0. One transaction bar-code is composed of multiple horizontal bar-code lines, where each horizontal line is made up of four cells. Among the four cells, the left most cell alternates black and white to indicate the change of line when DigitalSeal scans the lines. The top three and the bottom three lines of a transaction bar-code indicate the beginning and end of the transaction bar-code, respectively. Therefore, one transaction bar-code consists of 27 lines, and can hold 63 ($= 21 \times 3$) bit information, which will be used for representing transaction data.

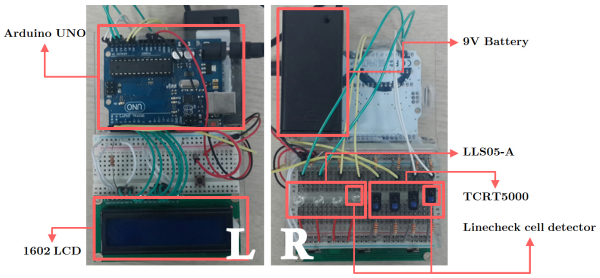

Fig. 1. DigitalSeal's prototype: front (L) and back (R).

3.2. DigitalSeal

As shown in Figure 1, a prototype of DigitalSeal was implemented using Arduino UNO [1], four LLS05-A sensors [7], four TCRT5000 sensors [9], a 1602 LCD and a 9V battery. Though the current prototype looks bulky, it can be easily manufactured in a more compact form (e.g., a thin bar with LCD display and light sensors) considering that its computation part requires only HOTP. Arduino UNO is a small computing device that can control various sensors, and it is easy use to build digital devices to interact with physical environments or objects. LLS05-A sensor can distinguish black and white colors displayed on a screen by using a light filter. The TCRT5000 sensor can distinguish black and white colors printed on a paper by detecting reflection of infrared rays. The 1602 LCD is for display, and 9V battery is used as a power supply.

The right side of figure 1 shows the layout of LLS05-A and TCRT5000 sensors. The rightmost among the four sensors, placed in the upper position, is the linecheck cell detector, and other sensors; the LLS05-A and TCRT5000. When a barcode is scrolled, the DigitalSeal detects first the change of lines, and then tries to read the data cells. If all sensors are aligned in a row, the data cell reading sensors try to read data as soon as the linecheck cell detector recognizes the change of lines. However, it is very likely to fail to read stably the data cells because the sensors are most likely to be in the border of two lines. Instead of aligning the four sensors in a row, we place the linecheck cell detector in a little upper position to recognize line change when data cell detectors are not at border of two lines, but in the middle of the data cells. Thus, we could obtain stable reading performance both for online and offline transactions.

3.3. Output Format of Data

The transaction input to DigitalSeal is encoded in ASCII, and the brief transaction information and HMAC are displayed

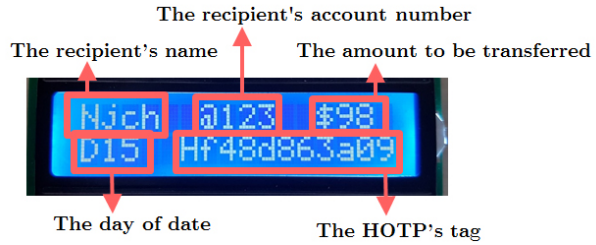

Fig. 2. Scanning results. Bar-code data and the HMAC's tag are displayed on the 1602 LCD.

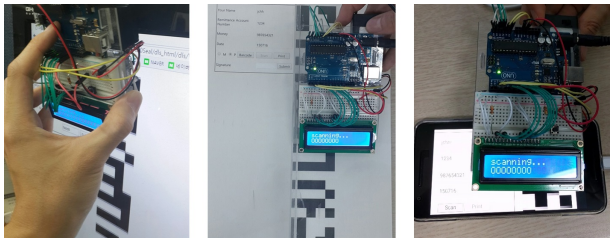

Fig. 3. Example of scanning a bar-code with DigitalSeal on the monitor, paper and smartphone using LLS05-A sensors and TCRT5000 sensors of DigitalSeal.

on DigitalSeal's LCD. The user checks whether the transaction information is correct or not. If correct, the user uses the HMAC as OTP for authorizing the transaction, and aborts otherswise. The current prototype uses three alphabetical characters and seven numbers to show the transaction information, and ten hexadigit code of HMAC(The HOTP's tag) for OTP as shown in Figure 2. The length of the truncated HMAC is recommended to be longer than 10 hexadigit numbers [19]. Thus, in reality, we can use 10 hexadigit code for OTP. Also, depending on the application of DigitalSeal, more information can be displayed (sender's names, time, etc.).

3.4. Using DigitalSeal

DigitalSeal handles multiple forms of authentication, both online and offline. To test scanning performance of DigitalSeal on a monitor and a smartphone, we made a webpage, on which a transaction bar-code goes up line by line every 0.1 second while users are placing DigitalSeal over the bar-code. The left snippets of Figure 3 shows an example of bar-code scanning with DigitalSeal on a computer monitor, while the right snippet show a scenario using smartphone. For scanning test on a paper, we made a scan auxiliary tool that helps users to scan a bar-code exactly and vertically on the paper. Our tool is made of a transparent acrylic panel to hold the bar-code paper and guide DigitalSeal to scan in a straight line, as shown in the middle snippet of Figure 3.

We envision that DigitalSeal will be issued by banks or government agencies, and will be loaded with secret key

Table 1. Result of testing session showing that most participant could use DigitalSeal well without any difficulty.

monitor	participants	10
avg	success rates / scanning times	98% / 2.6s
paper	participants	10
avg	success rates / scanning times	94% / 3.2s

shared with the issuer: when a bank issuing DigitalSeal shows an online transaction page, an IOU (I owe you) document, or a contract paper on which a bar-code is printed, a user scans the bar-code with his DigitalSeal device. After scanning, the bar-code data and the corresponding HOTP will be displayed on the 1602 LCD of DigitalSeal. The user then checks whether the displayed transaction information is correct or not. If the information is correct, the user inputs in an online page or writes on a paper the HOTP to sign the transaction. Figure 2 shows the result of scanning. The bar-code data (Njch, @123, $98, D15) and the HOTP's tag (Hf48d863a09, truncated HOTP) are printed out, where "jch" is the initial of the recipient name(N), "123" is the recipient's account number(@), "98" is the amount($) to be transferred, "15" is the day(D) of date, and the last 10 hexadigits(H) is the OTP.

4. USER STUDY

We conducted a user study with 10 participants for evaluating the usability of DigitalSeal by recording the success and failure rate and time taken for scanning. Our experiment was conducted to measure DigitalSeal's scanning performance on a desktop monitor and on a paper. Each participant were asked to scan five times the bar-code on the monitor and the paper respectively. For the test on the monitor, a webpage in which a transaction bar-code goes up line by line every 0.1 second was used, so the scanning times were all the same for all users. For testing on the paper, participants scanned from top to bottom of the transaction bar-code on the paper with the help of the scanning auxiliary tool. Table 1 shows the results of the testing sessions. For the test session, participants scanned five times the bar-code on the monitor and the paper respectively. For the monitor test, the success rate of 10 participants was 98%, and the average scanning time was 2.6 second. For the paper test, the success rate of the 10 participants was 94%, and the average scanning time was 3.2 second. As a result of the test sessions, we claim that DigitalSeal is usable to most users without introducing any difficulty.

5. APPLICATIONS

DigitalSeal generates the HOTP's tag from user's data by using a secret key shared to users. Therefore, users have to share the secret key in advance with the issuing institute. We assume that the issuing institute assigns a serial number and

embeds a secret key matched with the serial number in DigitalSeal, and also administer the secret key safely. The followings are examples of usage DigitalSeal on the online banking and IOU documents.

5.1. Online Transaction

First, a customer should request issuance of DigitalSeal in person to the issuing institute, e.g., bank. The bank authenticates the customer and his identifier (sign-in account name) through face-to-face authentication and issues DigitalSeal having a serial number to the customer. The bank saves the customer's identifier, the serial number and the secret key in a database safely. Customers do not need to know explicitly the secret key hard-coded in DigitalSeal. The customer signs in to the online banking site, and then it registers the serial number of DigitalSeal. Then, the bank server looks up the customer's identifier and the serial number in a database. If matched, the initialization is completed.

After initialization, a customer enters the recipient's name and account number, money, and date to transfer to other online banking accounts. The bank server makes and prints out the bar-code by using transaction information provided from customer on the monitor. When the customer scans the bar-code with DigitalSeal on the monitor, the brief transaction information and HOTP's tag are displayed on the 1602 LCD. After confirming the transaction information, the customer inputs the HOTP's tag on the bank's online transaction page. Upon receiving the transaction request with the HOTP's tag, the bank server calculates the corresponding HOTP's tag for the transaction and compares it with the received HOTP's tag and approves the transfer request if matched.

5.2. Offline Transaction

In this transaction, contractors (both a creditor and debtor) should first hold their own DigitalSeal issued by a trusted third party (e.g., the Department of Legal Services; or DLS for short). Registration of DigitalSeal to the website of DLS is same as that of the online banking transaction case.

The contractors get together to make a contract with DigitalSeal. Either the debtor and the creditor inputs the creditor's and debtor's names, account number, amount, day of date, and so on, in order to request to issue the IOU documents on the DLS site or some previously installed software that supports DigitalSeal. DLS server or the software makes the bar-code for the IOU information provided, and prints it out with the IOU information on a paper. Both the debtor and the creditor scan the bar-code on the IOU document with DigitalSeal, and the HMAC-based OTP of 10 hexadigits will be displayed on each DigitalSeal reader. They both confirm the IOU information displayed on the DigitalSeal screen. If they are matched, the debtor and the creditor write their own HOTP's tag on the IOU documents to approve and sign the deal. To validate the signed deal, both debtor and creditor simply can check on the DLS server by asking for the validity of the HOTP's tag.

6. SECURITY AND COST ANALYSIS

DigitalSeal is stronger than software solutions running on a PC or a smartphone in the sense that it is a hardware token separated from the transaction device. As such, an attacker does not have a route to penetrate the device. It is also strong against transaction fraud such as MitM [18] and MitB [4] attacks because its HOTP's tag is depends on transaction information. Therefore, an attacker cannot use an eavesdropped HOTP's tag for another purpose, which will make up the weak point of previous existing OTP. Also, it is strong against replay attack, because the transaction information has a date information and can include time information, or the server can challenge with a bar-code including a nonce. DigitalSeal can be used also for offline transaction like issuing and verification of IOU documents. It can reduce the forgeability of legal seal or signatures, both of which are static and do not change. DigitalSeal's HOTP's tag depends on the IOU document content and changes for every transaction. We can implement the DigitalSeal using a QR-code and camera component instead of the bar-code and some sensors. But we tried to make the DigitalSeal with the least expense because the DigitalSeal needs to be reasonably priced to be used in real life, if it is expensive, user will not use the DigitalSeal. So we designed a dedicated bar-code and implemented the DigitalSeal with LLS05-A sensors and TCRT5000 sensors. We expect that DigitalSeal can be implemented at a cost of about $1 USD when it is commercialized and manufactured with a mass production system (at scale).

7. CONCLUDING REMARKS

We introduced DigitalSeal, a transaction authentication tool that works in both online and offline use scenarios, and works in multiple settings, including scanning transaction information on paper, computer monitors, and mobile phones. We confirmed that DigitalSeal is usable and makes up the weak point of previous existing OTP. In the future work, we will further explore other usability of DigitalSeal, by large scale deployment, and ways to address unforeseen issues.

Acknowledgement

This research was supported by the Global Research Laboratory (GRL) Program of the National Research Foundation (NRF) funded by Ministry of Science, ICT (Information and Communication Technologies) and Future Planning (NRF-2016K1A1A2912757) and by NSF grant CNS-1809000.

8. REFERENCES

[1] *Arduino/Genuino UNO*. https://www.arduino.cc/en/Main/ArduinoBoardUno.

[2] *Challenge-response Authentication*. https://en.wikipedia.org/wiki/Challenge-response_authentication.

[3] *A Closer Look at Ezio TAN by Gemalto (GERMAN)*. https://www.youtube.com/watch?v=5oaaBNxW6b4.

[4] *Concepts against Man-in-the-Browser Attacks*. http://www2.futureware.at/svn/sourcerer/CAcert/SecureClient.pdf.

[5] *Defender software tokens*. https://software.dell.com/products/defender/softwaretokens.aspx.

[6] *Gemalto EZIO TAN Generator*. http://www.oberbank.de/OBK_webp/OBK/Informationsobjekte/Downloads/DE/ipkde_bedchipt.pdf.

[7] *LIGHT SENSOR*. http://en.nysenba.com/upfiles/file/lightsensor.pdf.

[8] *RSA SecurID*. https://en.wikipedia.org/wiki/RSA_SecurID.

[9] *TCRT5000, TCRT5000L*. http://www.vishay.com/docs/83760/tcrt5000.pdf.

[10] *Transaction authentication number*. https://en.wikipedia.org/wiki/Transaction_authentication_number.

[11] Mihir Bellare, Ran Canetti, and Hugo Krawczyk. Keying hash functions for message authentication. *Advances in Cryptology - CRYPTO '96*, 1109:1–15, 1996.

[12] Mihir Bellare, Joe Kilian, and Phillip Rogaway. The security of the cipher block chaining message authentication code. *Journal of Computer and System Sciences*, 61:362–399, 2000.

[13] F. Hoornaert D. Naccache O. Ranen D. M'Raihi, M. Bellare. Hotp: An hmac-based one-time password algorithm. *RFC4226*, 1997.

[14] M. Pei J. Rydell D. M'Raihi, S. Machani. Totp: Time-based one-time password algorithm. *RFC6238*, 1997.

[15] RhongHo Jang, Jeonil Kang, Aziz Mohaisen, and Dae-Hun Nyang. Rogue access point detector using characteristics of channel overlapping in 802.11n. In *37th IEEE International Conference on Distributed Computing Systems, ICDCS 2017*, pages 2515–2520, 2017.

[16] Byung-Tak Kang and Huy-Kang Kim. A study on the vulnerability of otp implementation by using mitm attack and reverse engineering. *Journal of the Korea Institute of Information Security and Cryptology*, 21(6):83–99, 2011.

[17] Jeonil Kang, DaeHun Nyang, and KyungHee Lee. Two-factor face authentication using matrix permutation transformation and a user password. *Inf. Sci.*, 269:1–20, 2014.

[18] Charlie Kaufman, Radia Perlman, and Mike Speciner. *Network security: private communication in a public world, second edition*. Prentice Hall Press Upper Saddle River, NJ, USA, 2002.

[19] H. Krawczyk, M. Bellare, and R. Canetti. Hmac: Keyed-hashing for message authentication. *RFC2104*, 1997.

[20] YoungJae Maeng, Aziz Mohaisen, Mun-Kyu Lee, and DaeHun Nyang. Transaction authentication using complementary colors. *Computers & Security*, 48:167–181, 2015.

[21] DaeHun Nyang, Aziz Mohaisen, and Jeonil Kang. Keylogging-resistant visual authentication protocols. *IEEE Trans. Mob. Comput.*, 13(11):2566–2579, 2014.