Multimed Tools Appl (2015) 74:3289–3303
DOI 10.1007/s11042-014-1888-3

Case study of the vulnerability of OTP implemented in internet banking systems of South Korea

Changsok Yoo · Byung-Tak Kang · Huy Kang Kim

Published online: 14 February 2014

Abstract The security risk of internet banking has increased rapidly as internet banking services have become commonly used by the public. Among the various security methods, OTP (one time password) is known as one of the strongest methods for enforcing security, and it is now widely used in internet banking services. However, attack methods which can detour OTP have been developed that additional security for OTP is now needed. In this study, we discovered that a new kind of attack through OTP is theoretically possible through an analysis of the currently implemented OTP system and known attack methods. Based on our theory, we tested the new attack method on Korean internet banking services, and empirically proved that it could effectively detour around all of the currently implemented OTP security systems in Korea. To prevent this, we also suggested solutions based on the root cause analysis of the OTP vulnerabilities.

Keywords OTP (one time password) · Man-in-the-middle attack · Reverse engineering · Internet banking

1 Introduction

The first internet banking service started in 1994 with 100 customers [30], but it has been rapidly grown that internet banking service is now known as one of the most frequently used internet services. The growth of money transactions via the internet has lured hackers seeking illegal capital gains. Unfortunately, most of the attacks by hackers have focused, not on the internet banking servers with tremendous security equipment, but on ordinary users to steal from their accounts.

At first, hackers preferred using malware or key loggers to steal users' inputs or certificate files. Careful users and ordinary personal security systems were sufficient to prevent these attacks. Nowadays, various social engineering techniques such as pharming, phishing, and smishing are combined with malware, which has made it difficult for ordinary users to avoid hackers' attacks.

C. Yoo
Department of Culture & Tourism Contents, Kyung Hee University, Seoul, South Korea
e-mail: csyoo@khu.ac.kr

B.-T. Kang · H. K. Kim (✉)
Graduate School of Information Security, Korea University, Seoul, South Korea
e-mail: cenda@korea.ac.kr

B.-T. Kang
e-mail: window31@korea.ac.kr

Hackers can even attack a banking service system itself using infected PCs, which has actually occurred in Korea. From March 20th to 26th in 2013, a cyber-terror targeting major banks in South Korea occurred. Although the hackers failed to access the core banking service system directly, but they massively destructed approximately 48,700 computerized equipment including PCs, servers, and network devices in the top four banks in Korea. This attack caused considerable harm to the reputations of these banks [5, 24].

All internet banking service providers are now trying their best to prevent account theft and keep users' PC clean. They apply various security mechanisms to protect the account information of customers. Among them, OTP (one time password) is now preferred and has been enthusiastically adopted as a two-factor authentication method [31]. Literally, OTP creates a new password whenever a user tries authentication, and then disposes of it instantly after use. This can effectively prevent the risk of password leaks [29]. Thus, hacking seldom occurs in accounts under OTP protection. Also, OTP can be generated by various devices or channels [3, 12, 17, 26]. Although generated OTP password value can be snatched by key sniffing, various study results confirmed that it is safe because the effective life-time of the snatched OTP is very short, usually about 60 s.

However, the risk of account theft still remains considerably due to the rapid evolution of hacking methods. To sustain the current security strength of OTP in internet banking, forecasting the future evolution of hacking, especially in relation to OTP, is definitely needed prior to actual attacks.

Therefore, this study will investigate the current security measures and hacking attack patterns in internet banking services, and theoretically forecast possible OTP attack scenarios. We will also discuss the results of a test that empirically demonstrated the effectiveness of our scenario. Based on the actual results, we will suggest a possible path for the future evolution of hacking in relation to internet banking and feasible solutions to protect against it.

2 Related studies and trends

2.1 Security safeguards

There are many security safeguards to protect users' accounts. They can be categorized in several ways. In this paper, from the data process perspective, we can categorize the security safeguards into five layers: the physical layer, network layer, OS/platform layer, database/application layer, and data layer.

Figure 1 summarizes the currently deployed security safeguards for protecting user authentication in internet banking services from the data process perspective. Each security safeguard has its own pros and cons, therefore multiple safeguards are usually installed from the viewpoint of defense-in-depth. Nonetheless, security safeguards working on users' PCs so far can easily be compromised by hackers because there are lots of vulnerabilities in the OS itself or running applications on OS. Therefore, strong user authentication methods are definitely needed to secure the transactions, especially web transaction security [9] between users and service providers.

There are also various user authentication techniques, and we divided them into four groups depending on the characteristics of authentication: simple password methods, second factor authentication methods, additional channel methods, and others. Table 1 summarized the known key characteristics of each authentication technique from the user convenience and security risk perspectives [13, 14, 18].

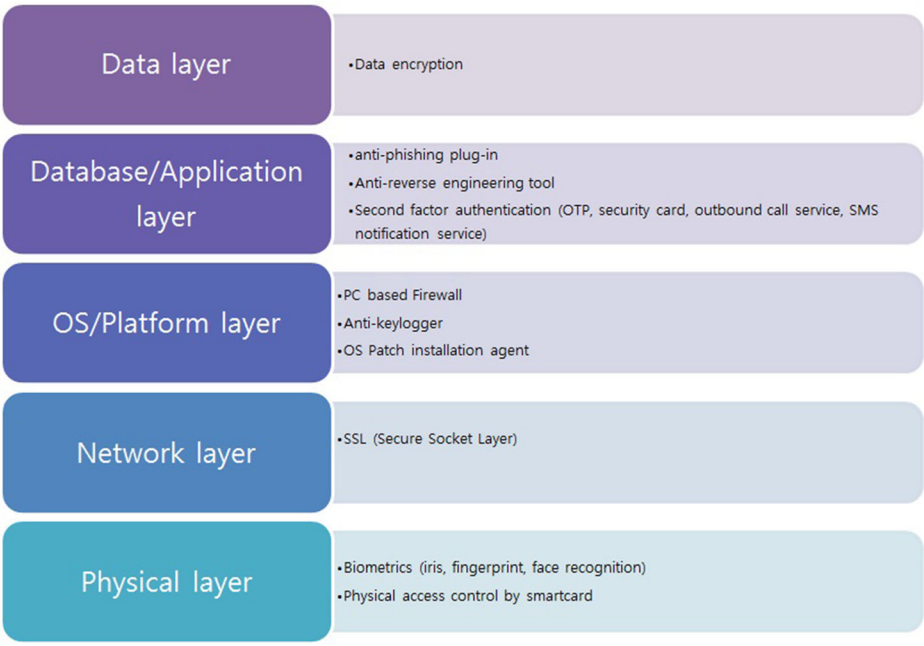

Fig. 1 Security safeguards implemented in Internet banking services

For decades, as like the malware target for internet banking services evolve, many security services are developed as countermeasure. The current status quo of security services of internet banking in South Korea is summarized by Lee and Kim [2]. In Table 1, we considered three major factors to measure user convenience. First, user inconvenience indicates any complexity in manipulation, movement, and operation. Users may experience inconvenience while using the authentication technique. Second, the installation cost denotes the complexity of the software or hardware installation and maintenance. Lastly, user carelessness denotes the expected problems that may occur due to carelessness of users in the authentication process.

From the security risk perspective, we can summarize the experts' estimations of the vulnerabilities in each authentication technique for the major attack patterns: sniffing, keyboard logging/screen capture, and phishing. If the authentication technique can counteract the attack pattern, it is marked with an O in the table. If not, it is marked with an X in the table. If the authentication technique can partially counteract the attack pattern, it is marked with a Δ.

There are five major authentication types in simple password method: ID password type, virtual keyboard, pre-inquiry response, security card, and image verification. The ID password type uses an ID and password as the primary authentication measure. It confirms the authentication by matching the inputted data with a pre-defined ID or password. The virtual keyboard type is almost identical to the ID password type, except for the input system. It uses a virtual keyboard rather than a physical keyboard to counteract keyboard logging. The pre-inquiry response type authentication asks for additional personal information that is already registered in the database along with the ID and password for the authentication. The security card and image verification type uses similar algorithms. They perform the challenge/response and application functions. CAPTCHA is a well-known application of image verification type authentication [25].

In the second factor methods, there are two major authentication types: symmetric key type and asymmetric key type. The symmetric key type prepares a secret key between a user and a

Table 1 Summary of user authentication techniques (O: Can counteract, X: Cannot counteract, Δ: Can counteract partially)

Class		User convenience			Security risk		
		User inconvenience	Installation cost	User carelessness	Sniffing	Keyboard logging/ screen capture	Phishing
Security safeguard for user authentication	ID password type	Low	Low	Low	X	X	X
	Virtual keyboard	Normal	Low	Low	O	Δ	X
	Pre-inquiry response type	Normal	Normal	Normal	X	O	X
	Security card	Normal	High	Normal	O	O	Δ
	Image verification type	Normal	Normal	Normal	O	X	Δ
	Asymmetric key type	Normal	High	Low	O	O	X
	Symmetric key type	Normal	High	Low	O	O	X
	One-way type	Low	High	Normal	O	O	X
	Two-way type	Normal	High	Normal	O	O	X
	Keyboard security	Normal	Low	Low	O	O	X

service provider, which is exclusive to a specific user. When communication occurs between a user and a service provider, the authentication system executes an encoding or message authentication code (MAC) operation using this key. OTP, the focus of this study, falls into this type. In the asymmetric key type, the authentication system distributes personal keys and open keys in pairs; if one party sends any content with his/her own personal key, the other party should check the content with the open key. The digital certificate is a major example of this.

In additional channel methods, there are two types of authentication: the one-way type and two-way type. The one-way type transfers a single-use password via a mobile communication terminal or other application program. After receiving the password, it performs the authentication through the internet browser. The two-way type is a slightly more sophisticated version of one-way method. Like one-way type, the password value is received via a separate channel. Before terminating the secured process, two-way type asks for the confirmation of the request via the separate channel. After receiving a response from the user, it finishes the request. Major examples include security safeguard such as a telephone approval service [21]. Nowadays, internet banking services have begun to provide notification services. When a user is logged into an internet banking site, the system notifies the login event to the user's cell phone. In addition, there has also been an increase in the adoption of E2E (End to End) function to complement the defect of keyboard security solutions [6]. Unlike keyboard security solutions which only protect the key board input to a browser, the E2E function is intended to protect the input value from the keyboard to the server by encoding. This is the one of the key methods to prevent sniffing in a network segment [4].

Various studies have evaluated the effectiveness and vulnerabilities of authentication techniques. We summarize these evaluations in Table 1, focusing on major attacks like sniffing, key logging/screen capture, and phishing [7, 10].

Although the ID password authentication type is the basic authentication method, it is vulnerable to most of major attacks [19]. The virtual keyboard type can effectively evade sniffing attacks, but is still vulnerable to malware which uses screen captures. The pre-inquiry response type involves so much information that it is easily presumed or exposed. Thus, social engineering questions such as "What is your father's name?" or brute force attacks can easily disarm the pre-inquiry response authentication. The security card or image verification method is relatively stronger than other simple password methods, but still is vulnerable to MITM (Man-in-the-Middle) attack.

2.2 OTP strengths and weaknesses

OTP is a kind of symmetric key type authentication which creates a one-time password whenever a user logs in. Synchronization between the OTP server and the OTP-creating-terminal is required to create the one-time password. There are three ways to maintain the compatibility between the OTP server and terminal: event-based synchronization, time-based synchronization and a hybrid type that combines both methods. But, time-based synchronization is commonly used in practice [23].

To create OTP, various transmission methods such as SMS transmission, mobile-phone applications, and hardware-type tokens are used. Thus, OTP can retain the attributes of two-factor authentication [3, 12, 17, 26].

Ordinary ID password authentication uses the same password value permanently that there is a strong possibility of account theft. To lower the risk of account theft, service providers recommend users to change their password frequently, but this does not prevent password leaks unless the change is made immediately after a hacker snatched the password. In this regard, OTP is free from password leaks because it disposes of passwords right after the use and does not create the same password again. Therefore, hacking has seldom occurred in accounts under OTP protection.

Although an OTP password value can be snatched by sniffing, various study results have confirmed that it is safe, despite the leakage, because the effective life time of a snatched OTP is very short, usually about 60 s [15]. However, internet banking service providers have found that there are frequent OTP input failures by users with the spread of OTP device. Although the effective life time range of OTP is about 60 s by default, older people or people that are unfamiliar with technology have difficulties to input OTP within the limited time because of their low typing speed or lack of understanding of OTP.

Thus, some internet banking service providers have increased the OTP life time, but this has only created security problems. Citibank experienced an OTP hacking incident because they set the OTP life time over 3 min for users' convenience. After they found the weakness in the OTP life time, they shortened it to 30 s [27]. Currently, 60 s of OTP life time has become the de-facto standard in South Korean banks.

2.3 Known OTP hacking method

Although OTP is one of the strongest authentication method to protect against account theft, there is always a possibility of account theft due to the evolution of hacking methods. To attack accounts under OTP protection, hackers developed MIT-X (Man-in-the-X) attacking methods. The reported attacking methods of MIT-X are summarized in Table 2.

The MITM attack (Man-in-the-Middle attack) accesses data transaction paths, and snatches information in the middle of data transactions [28]. The MITB attack (Man-in-the-Browser) refers hacking methods which use malignant programs in the web browser. The MITB attack

Table 2 Attack method comparison

Attacking method	Attack location	Example
MITM (Man-in-the-Middle)	Access path, access channel	Attacking network segment, Man-in-the-Middle attack
MITB (Man-in-the-Browser)	Web browser	Internet Explorer, BHO, JavaScript
MITPC (Man-in-the-PC)	PC environment	Keyboard, mouse logging, screen capture, API monitoring, code modification

induces users to enter important information such as their account number and password into a fake form. The MITPC attack (Man-in-the-PC attack) abuses the vulnerability of hardware environment or the operating system. It controls and attacks every data access path in the OS including the key input, mouse movement, screen, and even memories [22].

To attack OTP, hackers usually devise attack methods that combine several concepts in the MIT-X class.

2.3.1 OTP attack using MITM and phishing

An MITM attack combined with phishing is the most discussed format for an OTP attack. This theory works as follows. First, attackers intercept the certification through hacking a user's PC. They also find account information such as the user's ID and password using a phishing site. Then, when the user tries to log in using OTP, the attackers intercept OTP and perform an MITM attack.

In November 2005, Swedish Internet Bank temporarily closed their internet banking site because of a phishing attack. Citibank also reported cases of MITM attacks using a phishing site similar to the citibusiness service in July 2006 [8, 20]. The Symantec Institute reported the appearance of malignant code like Bank Silent, which makes internet banking service users vulnerable to account theft by MITM attack [1].

2.3.2 OTP attacks using MITB

MITB refers to hacking by installing a malicious program into the web browser of users. Unlike MITM attacks, it does not require a separate fake site. The main principle of this OTP attack is to overlay fake information input forms over the target internet banking service site using an installed malicious program. It is very hard for users to notice because it actually uses the internet banking service site. Malignant code for BHO (Browser Helper Object) of Internet Explorer or the overwriting of fake pages on a real site using JavaScript can be used to implement an MITB attack.

Unfortunately, most of the security solutions for users examine malignant codes which interrupt the actual transaction process, and do not protect or examine HTML document resources. Thus, MITB attacks have become an issue in dispute [16]. Augusto was the first who discovered the possibility of malignant code residing in a web browser. Guhring named this type of attack MITB [11].

2.3.3 OTP attacks using MITPC and MITM

After the reviewing current OTP attacks and security measures, we found a new kind of attack that combines MITPC (Main-in-the-PC) and MITM is theoretically possible. This method is

based on the MITM algorithm for account theft, but it intercepts the crucial account information not by a phishing site but by an MITPC attack.

The keyboard security solutions that internet banking service prefers can protect against information leaks from key input sniffing. But, theft attempt by code modification or API hooking (MITPC attack) using reverse engineering are hard to prevent using the current security solutions for internet banking services. After intercepting an OTP value using an MITM attack, hackers can immediately use OTP and previously gathered account information for account theft within the OTP life time.

3 OTP mechanism and basic hacking techniques

Before discussing our scenario combining MITPC and MITM, it would be better to explain how our scenario can work under the current OTP mechanism. The general OTP security procedure is as follows. First, when a user requests authentication, a new OTP is generated automatically in the OTP token or application. Then, the user sends this password to the authentication server to verify the validity of the OTP value. If the value input by the user is valid, the authentication is successful, and further services are provided to the user.

However, for OTP hacking, it is better to focus on the inter-process communication of the OTP information. The inter-process communication of OTP can be divided into five layers: keyboard, browser, wininet.dll, OS and network. When a user tries to transmit input values to the desired web service, the user have to input the transmitted OTP values using a keyboard. Using the values input by the keyboard, the web browser makes queries to transmit the input values to wininet.dll. Wininet.dll is a high level network library that Win32 provides. It supports HTTPS as well as HTTP and FTP, and has an interface for internet programming. In winininet.dll, the queries from the browser are changed into HTTP protocols through the HttpSendRequestA/W function, and are then transmitted through OS to the desired resources in the network.

In this process, the browser generates queries to wininet.dll using the HTTP POST/GET format, like name = notice&no = 337&otp = 123456. This raises two issues from the security perspective. The one is that the HTTP POST/GET format exposes not only the values but also the arguments. The other is that the communication between the browser and wininet.dll uses non-encrypted plain-text queries. Therefore, a hacker can not only easily monitor the queries communicated in the HTTP format but can also sniff the contents of OTP as well as the id or password, when the HttpSendRequestA/W of wininet.dll is hooked and monitored.

SSL (secure socket layer) or HTTPS can be applied to protect the communication, but they only cover communication between OS and the network. The inter-process communication with wininet.dll still occurs prior to the encryption process. The keyboard security also cannot be a solution because it only encrypts the communication between the keyboard and the browser.

To resolve this, encrypting queries between browser and wininet.dll is need. However, no matter how strongly protected the query is, the actual OTP value should be transferred within the memory register like eax, ebx, ecx, and edx. Once a value enters the memory register, it can be vulnerable to extraction through reverse engineering.

4 Scenario for snatching OTP

Table 3 shows the pros and cons of the current major security safeguards that can protect users input from various sniffing attacks. SSL is a very famous and widely spread standard that can

Table 3 Comparing safeguards to protect various sniffing type

	Keyboard security	SSL
Interception principle	Protection of input values, such as IDT hooking, etc.	Encryption of communication channels
Key sniffing	Protected	Not protected
Packet sniffing	Not protected	Protected
Query sniffing	Not protected	Not protected

encrypt all of the messages between web-browsers and web servers. It can protect the data in the middle of transmission on the network-side. However, it has a weakness in that it cannot provide protection from the client-side attacks such as key sniffing.

In South Korea, a keyboard security solution is mandatory when users connect to internet banking web sites. This program is usually launched via an ActiveX control forcefully installed by the Internet banking web site. This program runs on the client-side (in the users PCs). Therefore, it cannot provide full traffic encryption.

As summarized in Table 3, neither a keyboard security solution nor SSL can protect from the query sniffing type of attacks.

Internet banking services favor and recommend OTP as a reliable security safeguard for two reasons. The one is that OTP values that a user created are only for a single use, and cannot be used repeatedly. The other is that OTP values have a short valid period (normally 60 s or less). Even if an OTP value is exposed, a hacker can hardly succeed in logging in ahead of the user within the given time limit. However, if the theoretical vulnerabilities revealed in the previous chapter are used, OTP can be snatched and used under the 60 s time limit by query sniffing or code hooking.

To focus on the internet banking service, we developed a code hooking scenario because query sniffing can be easily blocked by the encryption of inter-process communication. As we already addressed, the risk of query sniffing is due to the non-encrypted inter-process communication between the web browser and wininet.dll. To get rid of this risk, encrypting is an easy, economical but powerful solution, and some advanced security systems for internet banking have already introduced it. Therefore, we will focus only on the OTP snatching situation when the inter-process communication is already encrypted.

4.1 Snatching OTP scenario using code hooking technique

Most internet banking services which allow real-time wire transfer are now effectively encrypting the inter-process communication using a certificate and banking security module. Therefore, this scenario assumes that the internet banking service uses four main security systems in the service: (1) keyboard input encrypting, (2) banking module code for inter-process encrypting, (3) a certificate for personal authentication, and (4) OTP.

To snatch the OTP values input at a user's PC, the PC should be infected with the malignant code that has four main functions: certificate hacking, intercepting OTP values, transmitting the snatched data, and securing the OTP life time.

4.1.1 Certificate hacking

To perform certificate hacking, the malignant code should perform two actions. The first action is to intercept the certificate password, and the second is to intercept the certificate file itself.

To obtain the certificate passwords, sniffing key entry is widely used. However, when keyboard input encryption system is applied in the user's PC; the code hooking technique should be applied. After reverse engineering the banking DLL, an export function, GetSecPassword() can be found. If the code is hooked there, the certificate's password is carried over as the first factor, and the certificate password can be obtained.

To intercept the certificate file, the common behaviors of certificate usage are the main key. Usually, internet banking services recommend users to store the certificate in a USB device. Therefore, the malignant code periodically inspects to see whether a USB device is inserted. If an external disk is detected, the malignant code searches for the certificate folder and transfers the folder to the hacker in real-time, along with the password intercepted a moment before.

4.1.2 Intercepting OTP value and account password

OTP values and account passwords should be transferred to the keyboard security DLL or internet banking DLL via registers, such as eax no matter how strong the security modules are. Therefore, code hooking method which was used in the certificate password hacking can also intercepts the OTP value and account information at the memory terminal.

4.1.3 Transmitting snatched data

After the account information is acquired, the id, password, and OTP values should be immediately transmitted to a hacker considering the short life time of OTP. To accomplish this more efficiently, a hacker's monitoring system usually uses an alarm when a snatched account is transmitted.

4.1.4 Securing OTP life time

If a user with snatched OTP information successfully authenticates prior to a hacker, the snatched OTP information immediately becomes invalid regardless of the OTP life time. To secure the OTP life time which is generally less than 60 s, a hacker should prevent the user's authentication. The prevention logic is simple. When a login attempt is detected, the hooking procedure changes the user's OTP input to false OTP input. After then, the malignant code transmits the false OTP input to the external network. This makes the user repeatedly receive a message about incorrect password until the end of the OTP life time. Even if the user recognizes the hacking attempt, the usual OTP life time (60 s) is too short for the user to effectively react. If the user attempts to change his password or call the banking security service after recognizing the hacking attempt, this would take more than one minute right after the successful hacking attempt.

4.1.5 MITM attack based on the snatched OTP value

If the malignant code effectively works on a user's PC, a hacker can instantly acquire the OTP value along with the account information and certificate. By using a real-time MITM attack, a hacker can successfully enter a victim's banking account, and transfer money to the hacker's account using the snatched OTP and account information in 60 s.

4.2 Differences from existing hacking methods

The OTP snatching scenario we suggested uses a technique applying MITM and MITPC, and it differs significantly from the known MITM or MITB hacking methods in various ways.

4.2.1 Comparison with MITM attack

In the MITM attack, a phishing site is definitely required under any circumstances to secure the OTP life time. However, our scenario suggests that a phishing site is not mandatory, and an actual site can be used as an alternative. This will greatly increase the success rate of the attack, and also the security risk in the OTP system.

4.2.2 Comparison with MITB attack

Considering the use of an actual site, our scenario is close to the MITB attack. Although the MITB attack retrieves an OTP value from the actual site, it needs to use a fake web page like BHO on the actual site. Therefore, sensitive users can visually detect the hacking attempt. Moreover, anti-phishing solutions, which are now widely used, can detect malignant BHO or HTML pages, which are unnecessarily placed on the web page. In contrast, the attack scenario of this study does not modify a document of an actual page, but only focuses on the inter-process communication, which is hard to detect visually or by anti-phishing solutions.

5 Tests and results

To test our MITM attack scenario for internet banking services, we targeted internet banking services supporting OTP security in Korea, and tested the attack scenario from October to December in 2011. There are two reasons that we targeted Korean internet banking services. The one is the real-time wire-transfer service environment in Korea. To perpetrate an OTP attack with MITM and MITPC, the bank should support real-time wire-transfer services. Although most international banks support internet banking services, Korean banks are the only ones that provide a real-time transferring system for inter-bank transactions. Thanks to the strong investment drive of the government, all Korean bank support real-time wire-transfers among domestic banks, whereas the other countries still do not support this kind of real-time transfer between different banks. Ironically, the most advanced and convenient internet banking services in Korea now allow a new form of OTP vulnerability.

The other reason for targeting Korean banks is the security responsibility or control ownership concept of internet banking services. For most U.S. banks, the responsibility of banking service security falls not on the bank itself, but on the service users. Therefore, U.S., banks do not usually provide sophisticated security systems to evade malignant codes, but users need to maintain security of their PCs by themselves. However, in Korea, the responsibility of internet service security falls on the service providers. Thus, banks need to invest and develop the various security systems from keyboard encryption, and certificate authentication to OTP and bank security modules, which even cover encryption of the inter-process communication.

We targeted ten major Korean banks which allowed inter-bank real-time wire transfer and also provided OTP as a two-factor authentication during the experiment period.

For the experiment, we prepared two PCs: one for emulating a victim's PC and the other for a hacker's. We designated different IPs for the two PCs to make the test realistic. In the victim's PC, the malignant code for OTP snatching was installed. The malignant code was developed based on our scenario using reverse-engineering of the security modules of the 10 major banks. Bank accounts for the victims and a hacker were prepared using the authors' personal information. To simplify the test, we prepared one account per bank for victims. For the hacker's account, we used existing personal account of one of the authors for the test.

However, we used the different OTP tokens and certificates provided by each target bank. All the passwords for the bank accounts and certificates were generated randomly after reverse-engineering of the internet banking DLL.

To check the efficiency of the OTP's life time, the victims repeatedly attempted to log-in during the OTP life time, and the hacker also tried to transfer money from the victims' accounts to the hacker's account during the OTP life time, which was 60 s. Actually, most of the internet banking sites allow more than 60 s because not all internet banking users can input the OTP token value within the time limit. Therefore, values of 120~180 s are widely used for this reason, whereas 60 s is known to be the base lifetime of the issued token.

The experimental attack was performed during December of 2011, and we succeeded in accessing the victims' accounts and transferring money to the hacker's account using the snatched OTP and account information from all the target banks under 60 s of snatching the information. The detailed experiment results are summarized in Table 4.

6 Solution measures

The OTP snatching scenario suggested in this study is not based on the vulnerability of the OTP algorithm itself but that of conventional problems when implementing OTP in a windows system environment. In addition, most of the vulnerability comes from the real-time wire transfer environment used by Korean banks. Because of this, one can say that the vulnerability is just a local issue in Korea. However, considering the growth of the IT infrastructures of banks and the growing needs of customers for better service, real-time inter-bank wire transfers will be common in the near future. Therefore, we approached the solution development from two perspectives: the technology perspective, and the policy perspective.

6.1 Technological solutions

Because the suggested OTP snatching scenario mixed various hacking techniques, it is almost impossible to derive an ultimate solution for the scenario. Therefore, we suggest four solutions that can each eliminate one of the known vulnerabilities of the current system. To defend the entire system from the threat of a mixed MITM and MITPC attack, the four solutions should be mixed, and applied systemically.

Table 4 Test results for internet banking system in South Korea

Company	Keyboard security	Query sniffing or code hooking	Certificate hijack and authentication from USB	Reuse OTP
A	application	enable	enable	enable
B	application	enable	enable	enable
C	application	enable	enable	enable
D	application	enable	enable	enable
E	application	enable	enable	enable
F	application	enable	enable	enable
H	application	enable	enable	enable
G	application	enable	enable	enable
H	application	enable	enable	enable
I	application	enable	enable	enable

6.1.1 Mutual certifying in server

One of the strong boundary conditions against an attack on the current OTP systems is the OTP life time. Inevitably, when performing an attack in less than 60 s, the victim's IP and hacker's IP differ physically. Therefore, multiple log-in requests from different IPs within 60 s can be regarded as an attack attempt. Using this information, authentication servers can easily define whether a user's account was hacked, and block the transfer request immediately. However, there is always a possibility that an IP can be detoured, depending on the OTP system.

6.1.2 Protection of certificate

Currently, the certificate can be copied and used without a separate authentication procedure for the users' convenience, but this is one of the key vulnerabilities of current OTP system in our attack scenario. Interaction between the hardware serial information of the USB or PC and the certificate key or encoding key would prevent a hacker from using the transferred certificate on the hacker's PC. Therefore, a hardware certifying function should be included in the certificate for an internet banking service, especially for real-time transfer.

6.1.3 E2E (End to End Encryption)

Generally, a web browser and internet banking module use different encryption systems, which provides an opportunity for a hacker to infiltrate. E2E allows direct encryption from the web browser to the authentication server, which can effectively reduce the vulnerability in inter-process communication. However, E2E only covers the inter-process communication from the web browser to the network. Thus, E2E cannot eliminate the vulnerability in the key board input encryption system.

6.1.4 Code integrity checking

In our OTP snatching scenario, the hacker had to use the modified internet banking DLL to apply the code hooking technique, but the certification modules of the internet banking services did not perform a code integrity check before the authentication process. Although a code integrity check cannot be the ultimate solution due to the possibility of detouring by reverse engineering, it application could enhance the OTP security level at a low cost.

6.2 Policy perspective solutions

No matter how well prepared the security is, there are always vulnerabilities that a hacker can use. In that regard, technology cannot be the ultimate solution, and flexible policies should be followed to minimize the loss and risk. When a money transaction is executed in real-time, it is hard to roll back or retrieve the transferred money.

Criminals have found this vulnerability and abused it using telephone phishing techniques and social engineering in Korea. Criminals have called parents and told them that they have already kidnapped their child, and demanded a ransom for the safe return. They threatened to harm the child within several minutes if the parents did not transfer the ransom. If the shocked parent was deceived by the threat and sent the ransom, the criminals immediately withdrew the money using an ATM machine. This kind of telephone phishing became so popular that banks finally realized that the problem was the real-time transaction. To remedy this, Korean banks set up a transaction delay of 10 min for transactions greater than a thousand dollar.

This is an effective way to respond not only to phishing but to various MITM attacks. However, when a user's PC is fully compromised by a hacker, this transaction delay will not work as intended. Thus, we suggest a transaction notification system for transactions over a certain amount of money. This can serve as an alarm for the accountholder to identify the hacking attempt. The alarm methods include sending SMS to the cellphone of the account holder, or an outbound call when the transaction is suspicious.

7 Conclusion

In this study, we investigated the security measures for internet bank security, and suggested a hacking method for bypassing OTP using the MITM and MITPC techniques which can easily be developed in the near future. The significance of this study lies in the discovery that most internet banking services that are planning to implement real-time money transactions are exposed to the high risk of OTP bypassing by our scenario. To prove this, we performed tests on actual internet bank services that support real-time inter-bank money transactions, and found that the current security systems used by internet bank services did not cover our attack scenario.

The suggested scenario and experiment were designed to let everyone know that OTP snatching is possible. However, many internet banking services have failed to notice that OTP, as well as a customer's id, and password, can be obtained using a code hooking technique with DLL injection and reverse engineering. Thus, just implementing key board input encryption and HTTPS is not enough to sustain the security levels required for internet bank services.

Considering the root of the vulnerability in OTP, we suggested technical solutions which can effectively cover the security holes when implementing OTP in a windows system environment, and also suggested new policies to set up safeguards for real-time money transaction. Thankfully, our alert on Korean bank security has motivated banks and the government to adopt our solutions, but we strongly recommend that other banks that are now implementing and planning real-time wire transfer services examine their vulnerability to a mixed MITM and MITPC techniques. This will help internet bank services to operate more safely.

Acknowledgments This work was supported by a grant from the Kyung Hee University in 2013 (KHU-2013-0988).

A preliminary version of this paper appeared in "A study on the vulnerability of OTP implementation by using MITM attack and reverse engineering", Kang, Byung-Tak and Kim, Huy-Kang, Journal of the Korea Institute of Information Security and Cryptology, volume 21, issue 6, 2011, pp. 83–99. This version has been considerably improved from the previous version by including new results and features.

References

1. (2008) NetworkWorld, New Trojan intercepts online banking information, http://www.networkworld.com/news/2008/011408-silentbanker-trojan.html
2. (2014) Gi Seong Lee, Huy Kang Kim, "Internet Banking Security Services in South Korea, the status quo", http://www.hksecurity.net/internet-banking-in-south-korea
3. Aloul F, Zahidi S, Wassim E-H (2009) Two factor authentication using mobile phones. IEEE/ACS International Conference on Computer Systems and Applications, pp 641–644
4. Bae G, Lim G (2008) Analysis of basic weakness of keyboard security solution, Korea Institute of Information Security Cryptology, No.3, Vol. 18, pp 89–95
5. BBC News, South Korea blames North for bank and TV cyber-attacks, http://www.bbc.co.uk/news/technology-22092051
6. Chang H (2011) The study on end-to-end security for ubiquitous commerce. J Supercomput 55(2):228–245
7. Christos K (2007) Dimitriadis, analyzing the security of internet banking authentication mechanisms. Inf Syst Control J 3:1–8

8. Citibank Phish Spoofs 2-Factor Authentication, http://blog.washingtonpost.com/securityfix/2006/07/citibank_phish_spoofs_2factor_1.html
9. Considerations for web transaction security, RFC2084, http://www.ietf.org/rfc/rfc2084.txt
10. Cornel de Jong, Online authentication methods, evaluate the strength of online authentication methods, http://staff.science.uva.nl/~delaat/rp/2007-2008/p30/report.pdf
11. Guhring P (2007) Concepts against man-in-the-browser attacks
12. Hallsteinsen S, Jorstad I, Thanh D (2007) Using the mobile phone as s security token for unified authentication. In: ICSNC 2007. IEEE Computer Society, Los Alamitos pp 68
13. Hanacek P, Malinka K, Schafer J (2009) E-Banking Security—Comparative Study, 10th ACIS, pp 263–26
14. Hiltgen A, Kramp T, Weigold T (2006) Secure Internet Banking Authentication, IEEE Security & Privacy
15. Ku WC, Tasi HC, Tsaur MJ (2005) Stolen-verifier attack on an efficient smartcard-based one-time password authentication scheme. IEICE Trans Commun E87-B(8):2374–2376
16. Maeng Y, Shin D, Kim S, Yang D, Lee M (2010) Analysis of weakness of MITB against credit transfer of domestic internet banking, Internet and Information Security, No.2, Vol.1, pp 101–118
17. Mizuno S, Yamada K, Takahashi K (2005) Authentication using multiple communication channels, in DIM 2005: Proceedings of the 2005 workshop on Digital identity management. New York, NY, USA: ACM, pp 54–62
18. Oppliger R, Rytz R, Holderegger T (2009) eSecurity Technol, Internet Banking: Client-Side Attacks and Protection Mechanisms, IEEE, Computer, pp 27–33
19. Paulson LD (2002) Key snooping technology causes controversy, IEEE, Computer, pp 27
20. Phishing attack targets one-time passwords—scratch it and weep, http://www.theregister.co.uk/2005/10/12/outlaw_phishing/
21. Phone approval service, http://bank1.kbstar.com/quics?asfilecode=5023&_nextPage=page=B002346
22. Security aspects of the SuisseID - http://postsuisseid.ch/en/suisseid/security/security-aspects
23. Seo S, Kang W, (2007) Technical status of OTP & cases of introducing OTP in domestic financial institutions, Korea Institute of Information Security Cryptology, No.3, Vol. 17, pp 18–25
24. Sherstobitoff R, Liba I, Walter J (2013) Dissecting Operation Troy: Cyberespionage in South Korea, http://www.mcafee.com/au/resources/white-papers/wp-dissecting-operation-troy.pdf
25. Steeves DJ, Snyder MW (2005) Secure online transaction using a CAPTCHA image as a watermark, U.S.Patent, 11/157,336
26. Thanh D, Jonvik T, Feng B, Thuan D, Jorstad I (2008) Simple strong authentication for internet applications using mobile phones. IEEE GLOBECOM pp 1–5
27. UOTP, http://www.u-otp.co.kr/blog/
28. Wikipedia.: Man-in-the-middle Attack, http://en.wikipedia.org/wiki/Man_in_the_middle_attack
29. Wikipedia.: One-Time Password, http://en.wikipedia.org/wiki/One-time_password
30. Wikipedia.: Online Banking, http://en.wikipedia.org/wiki/Online_banking
31. Wikipedia.: Two-factor Authentication, http://en.wikipidia.org/wiki/two-factor_authentication

Changsok Yoo received the Master's degree in Engineering from Seoul National University in 2001, and the Ph.D. degree in Energy Economics from Seoul National University in 2011. He is currently an assistant professor in Kyung Hee University. His current research interests are in the area of online game economics, industry analysis, and information security from the context of security usability. Contact him at csyoo@khu.ac.kr.

Byung-Tak Kang is a master course student in Graduate School of Information Security, Korea University. He is also a team manager of information security team in Nexon America. His research interests include Online Game Security, network and system security, reverse engineering and anti-reverse engineering techniques. Contact him at window31@korea.ac.kr.

Huy Kang Kim received his Ph.D. in Industrial and Systems Engineering from Korea Advanced Institute of Science and Technology (KAIST) in 2009. Currently he is an assistant professor in Graduate School of Information Security, Center for Information Security Technologies (CIST) in Korea University. His research interests include Botnet Detection, Intrusion Detection System, Network Forensics and Online Game Security. Contact him at cenda@korea.ac.kr.