Int. J. Advance Soft Compu. Appl, Vol. 6, No. 1, March 2014
ISSN 2074-8523; Copyright © SCRG Publication, 2014

Prevention Schemes Against Phishing Attacks on Internet Banking Systems

Seoung Yeop Na, Hyun Kim and Dong Hoon Lee

Graduate School of Information Security, Korea University
e-mail: sy_na@korea.ac.kr
Graduate School of Information Security, Korea University
e-mail: bkcloud18@gmail.com
Graduate School of Information Security, Korea University
e-mail: donghlee@korea.ac.kr

Abstract

With the rise of Internet banking, phishing has become a major problem in online banking systems. Over time, highly evolved phishing attacks, such as active phishing, have emerged as a serious issue. Thus, we suggest two server authentication schemes based on SSL/TLS to protect Internet banking customers from phishing attacks. The first scheme uses the X.509 client certificate, which includes a personal identification message from the customer in order to recognize a genuine banking server. The second scheme, based on the first one, is a modified version of SSL/TLS. We also analyze our schemes using attack scenarios and an analysis table.

Keywords: *active phishing, attack tree, Internet banking, phishing, X.509 client certificate*

1 Introduction

With the remarkable surge in Internet banking, phishing has become a significant concern in banking systems. Data represented in Fig.1 shows that about 77% of all phishing attacks in the first half of 2013 targeted the financial sector. Phishing means that an attacker intercepts an Internet user's private information (e.g. ID, password etc.) using social engineering attack or concealment techniques. If an MITM (man-in-the-middle) attack is integrated into a phishing attack, we call this 'active phishing attack.' In an active phishing attack, an attacker forms independent connections with two victims – the client and the financial institution in the cases under consideration – and relays messages between them, tricking them into thinking that they are communicating privately with each other. In fact, however, the attacker is able to intercept any message from either end and replace

it. Although the SSL/TLS security protocol used by most Internet banking servers provides server-client interactive authentication and encryption communication, it can be vulnerable to active phishing attacks.

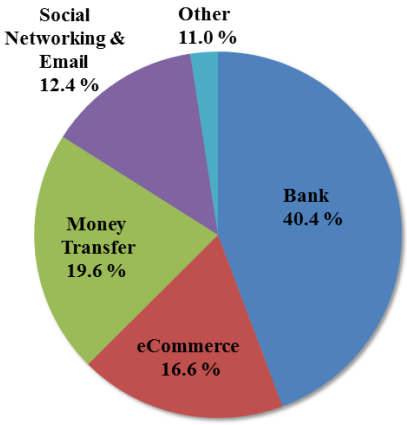

Fig.1 Industries attacked by phishing in the first half of 2013[4]

Fig.2 depicts an active phishing attack on a system using SSL/TLS. According to this scenario, an Internet banking customer's client (e.g. the web browser) is connected to an Internet banking server via an attacker's proxy server. Although the communication protocol has changed from HTTPS to HTTP on account of the attacker, as shown in the diagram below, most users do not realize this [17]. Thus, the unsuspecting customer proceeds to enter confidential information on the web browser and the unencrypted information is sent to the attacker [15].

Fig.2 Active phishing attack on SSL/TLS

To prevent phishing attacks, the customer has to be certain that she is connecting to a genuine Internet banking server. In this paper, we propose two server authentication schemes that can be applied to an Internet banking system. Our first scheme is based on SSL/TLS and uses Personal Identification Message (PIM) for customers to intuitively identify the genuine server. Our second scheme is founded on the first one, with an extra step to prevent active phishing attack.

This paper is organized as follows. Section 2 describes the related works. Section 3 illustrates an attack model and scenario in Internet banking systems. Section 4 introduces design requirements, while we present our proposed schemes in section 5. Section 6 contains an analysis of our schemes, and we reflect on our proposals in the concluding section.

2 Related Works

In this chapter, we introduce current server authentication techniques and limitations of those.

Fig.3 Example of EV SSL and SSL [9]

2.1 EV SSL

EV SSL (Extended Validation Secure Socket Layer) provides server-client interactive authentication and encryption. It also makes it easier for the user to verify an authenticated server because the EV certificate used in EV SSL provides additional information, including company location, corporate name and a registration number for the server website[5]. In Fig. 3, the images at the top and the bottom are instances, respectively, of EV SSL and general SSL. Users can easily recognize that a server provides EV SSL by noticing that the address bar turns green, as well as through the appearance of a padlock image and the name of the company in question. Thus, EV SSL is more intuitive than general SSL, and is currently being used online by most financial institutions.

However, a joint test by Harvard University and MIT in 2006 showed that only 9% of test participants were able to distinguish between a website using EV SSL and one using general SSL [17]. Furthermore, in another study by Stanford University and the Microsoft Corporation, most test participants could not distinguish a phishing website from the real one even when the website had been using EV SSL [6]. These results indicate that most users do not notice the changes in the appearance of the address bar of their browsers that indicate the two different security protocols. Thus, a banking server using EV SSL can also be vulnerable to active phishing attacks [15].

2.2 User-Selected-Image

User-selected-image is a phishing prevention technique that uses a personal image chosen by the online user. Bank of America's SiteKey is a representative example of user-selected-image [18]. It saves the customer's private image in a secure cookie. Whenever the customer connects to a Bank of America server, the server extracts the private image from the secure cookie and shows the image to the customer on a web page. If the customer cannot view this private image, she can conclude that the website in question is a phishing site. However, this technique

has a limitation, namely that the customer can only use the SiteKey on the computer which was originally used to create the private image [1].

To solve this problem, another SiteKey technique has been proposed. The customer enters her ID on the Bank of America web page, following which a bank server checks the validity of the ID and sends the customer her private image. If the customer recognizes the private image as the one she had chosen, she enters her password. However, this technique is also vulnerable to active phishing, as shown by the following scenario. First, an attacker who has a phishing website tricks the customer into entering her ID on the fake web page. Then, the attacker sends the ID to the actual server of the financial institution in question. The server responds by sending the private image associated with the ID to the attacker. Finally, the attacker sends the private image to the customer, who proceeds to enter her password under the impression that she is communicating with her financial institution. Hence, the attacker obtains the customer's password [1].

2.3 Phishing Prevention Solution based on Black-list(PPSBL)

A phishing prevention solution based on black-list(PPSBL) involves maintaining a list of known phishing websites on the customer's computer. If the customer accesses a website that is on the phishing site list, the solution will restrict the access to it. Instances of PPSBL are Google Chrome's Safe Browsing, Microsoft Internet Explorer's Phishing Filter and the Phishing Protection used by Mozilla Firefox [8,10,13].
According to survey [2], in 2011, the average lifetime of phishing sites was 46 hours while 50,298 phishing domains were identified.. This average lifetime decreased to 26 hours in 2012, while the number of phishing domains rose to 89,748[3]. This data shows that on account of the continuing rise in their numbers, phishing attacks are difficult to prevent through black lists. This is because phishing site experts have to manually update these lists and users only download these periodically[1]. Hence, users remain susceptible to phishing attacks from newly created sites that have not yet been detected

3 Attack model and scenarios

An attack tree model of Internet banking systems is presented in Fig.4. This model is based on an attack tree model[16] and represents various ways to steal customers' private information. According to the attack tree model, it is possible that an attacker exploit phishing attacks using a fake web page and a fake proxy server to obtain online access to a customer's account.

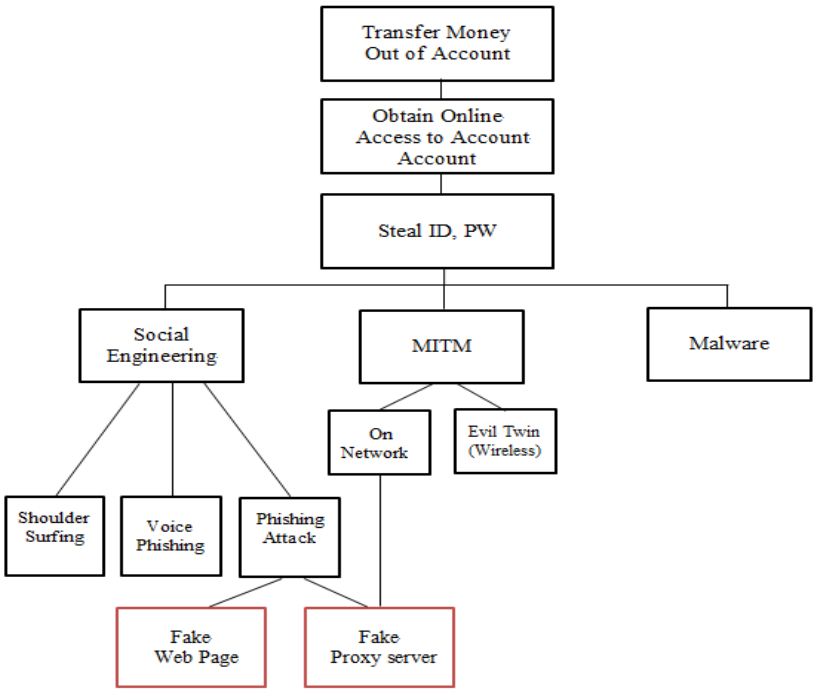

Fig.4 Attack tree model

3.1 Phishing attack

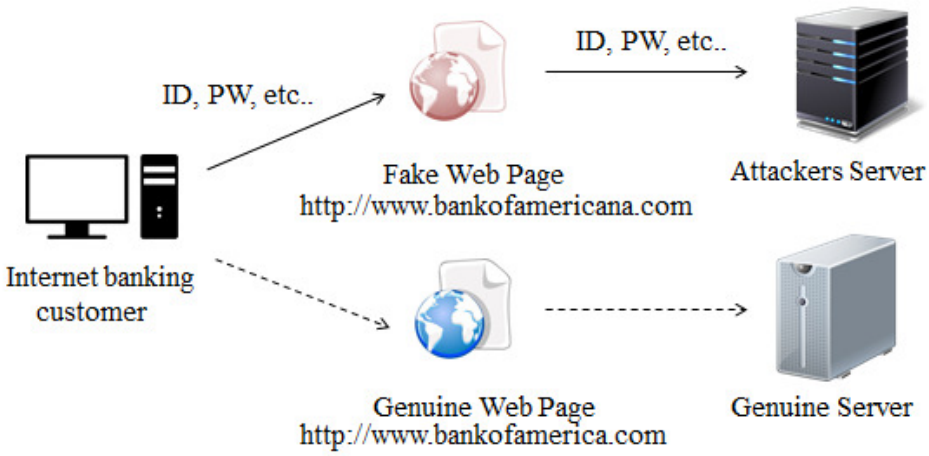

Fig.5 Phishing attack

A phishing attack is represented in Fig.5 and occurs as follows:

1. An attacker uses phishing emails or phishing links to lure an Internet banking customer into a fake banking web page[12].
2. The customer connects to the phishing banking web page, the web address for which is sufficiently similar to that for the actual website of the bank

in question to deceive an unsuspecting user. General phishing URL types are listed in [19].

3. The customer enters her personal information on the phishing site to log in.
4. The attacker steals the customer's information, accesses the customer's account and transfers the money elsewhere.

3.2 Active phishing attack

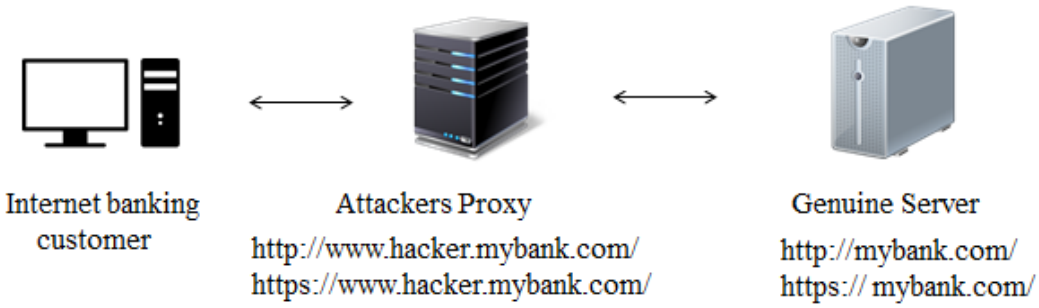

Fig.6 Active phishing attack[11]

An active phishing attack is depicted in Fig.6 and is as follows:

1. By using Transparent Proxies, URL Obfuscation or Browser Proxy Configuration, an attacker's proxy server can mediate between an Internet banking customer and a banking server[11].
2. The unwitting customer connects to the (attacker's) proxy server instead of the bank's server. At the same time, the attacker connects to the bank's server posing as the customer[11]. The proxy server is then able to relay web pages from the bank's server to the customer. If the bank's server uses the SSL/TLS protocol, the proxy server is also able to establish SSL/TLS in order to communicate with the customer as well as the bank's server.
3. The attacker proxies all communication and steals the customer's personal information.
4. Sometimes, Internet banking servers require additional authentication, such as two- channel authentication or one-time password. However, in an active phishing scenario, the attacker can succeed the additional authentication by relaying authentication value which was came from the customer to the genuine server.
5. The attacker accesses the customer's bank account and transfers money out of it.

4 Design requirements

It is very important that an Internet banking customer is able to authenticate an Internet banking server because failing this, the customer is vulnerable to phishing attacks. According to [14], the requirements of web authentication technique are usability, deployability and security. According to [1], requirements of defense to active phishing attack are usability, performance, security.

In light of aforementioned two requirements, we made requirements of phishing and active phishing attack. Those are usability, deployability and security. Usability and deployability are based on [14] and security is based on [1]. We will use these as criteria in section 6 for the analysis of our proposed schemes.

5 Our schemes

We first introduce the notations in our schemes. *PIM* (personal identification message) is a indication for an Internet banking customer to identify an Internet banking server; *EPIM* is encrypted *PIM,* a PIM encoded by a by k where k is secret key of the server. *SURL* is the known Server URL of the bank, while *URL* is the URL of a server to which the customer is connecting. $D_K(\cdot)$ is a decryption algorithm using the secret key k. $Cert_c$ is a X.509 client certificate that includes *EPIM* or *SURL*. $Cert_s$ is a X.509 server certificate. Lastly, w is a login web page.

5.1 SAPIM (Server Authentication using Personal Identification Message)

Our proposed server authentication scheme is based on SSL/TLS and uses a X.509 client certificate. We assume that the X.509 client certificate is mandatory for client authentication and is issued by an Internet server with which a customer wants to communicate or transact. When issuing a customer's client certificate, the *PIM* is encrypted by k and saved into the extension field of the client certificate. This *EPIM* can only be decrypted by k, the value of which is stored in the server of the financial institution. The *EPIM* is thus only decryptable by the actual bank server. Hence, the customer can be assured that she is communicating with the bank's server if she can recognize the decrypted *PIM*. Hence, SAPIM is composed of a certificate issuing process and a server authentication process.

5.1.1 Certificate Issuing

X.509 v3 is the standard of certificates in SSL/TLS. An extension field is an optional component of the certificate, and it consists of several components in turn. Among these, the Subject Directory Attributes component is used to confirm the identity of the certificate. The size of the X.509 certificate component is unlimited [7].
Client certificate issuing occurs as follows:

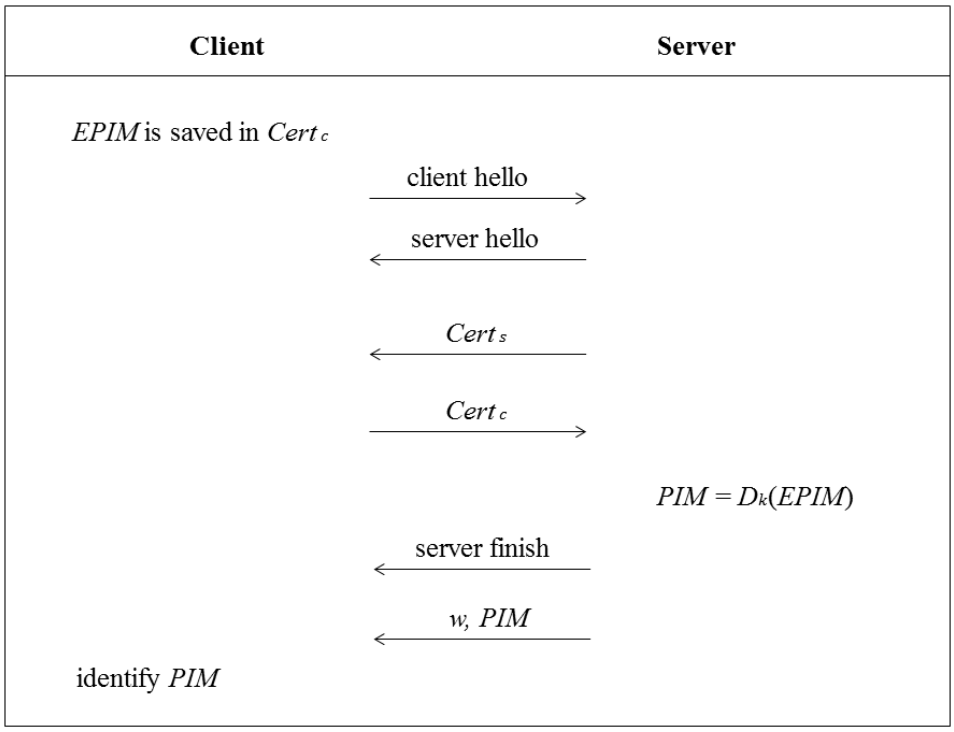

Fig.7 SAPIM

1. An Internet banking customer connects to a banking server and chooses a PIM.
2. The *PIM* is encrypted by *k* and saved in the Subject Directory Attributes of the customer's X.509 client certificate.
3. The server issues the $Cert_c$.

When $Cert_c$ is issued, we can be assured that the customer is communicating with a genuine Internet banking server through a secure channel. The *PIM*, which is a private phrase or image identifiable by the customer, is used to recognize the banking server.

5.1.2 Server authentication

Fig.7 shows the server authentication process of SAPIM, which occurs as follows:

1. The server and customer commence mutual authentication.
2. The server receives $Cert_c$ from the customer and decrypts the *EPIM* using *k*.
3. The SSL/TLS session is established successfully.
4. The server sends *w* and the *PIM* to the customer.
5. The customer receives the *w* and recognizes the *PIM*.

Using SAPIM in Internet banking systems, the customer can identify a genuine server more intuitively than in the current SSL/TLS protocol. A generic login page for SAPIM is shown in Fig.8 and Fig.9. The customer can recognize the PIM more easily than the changes in the appearance of the address bar of the web

browser, as in SSL/TLS. Furthermore, since SAPIM is based on SSL/TLS, any Internet server using SSL/TLS (e.g. Google) can use our SAPIM directly.

Fig.8 Example of a login web page using SAPIM(1)

Fig.9 Example of a login web page using SAPIM(2)

5.2 Advanced-SAPIM

SAPIM based on SSL/TLS is also not completely immune to an active phishing attack. This is because the SSL/TLS protocol ultimately relies on the customer's ability recognizing a genuine bank server. Thus, we suggest Advanced-SAPIM to resolve this issue. Advanced-SAPIM differs from SAPIM in that *SURL* is also saved in the client certificate in the former, along with an additional step. In this step, the client (e.g. web browser) checks to see if *URL* is identical to *SURL* saved in the customer's client certificate. Using this step, the customer is able to communicate with the preselected bank server instead of a malicious server.

5.2.1 Certificate Issuing

Advanced-SAPIM uses the identity of the Internet banking server to prevent active phishing. Therefore, *SURL* is saved in the client certificate. The client certificate issuing is as follows:

1. The customer connects to a banking server and chooses a PIM.
2. The PIM is encrypted by k, following which *EPIM* and *SURL* are saved in the Subject Directory Attributes of the customer's client certificate.
3. The server issues $Cert_C$.

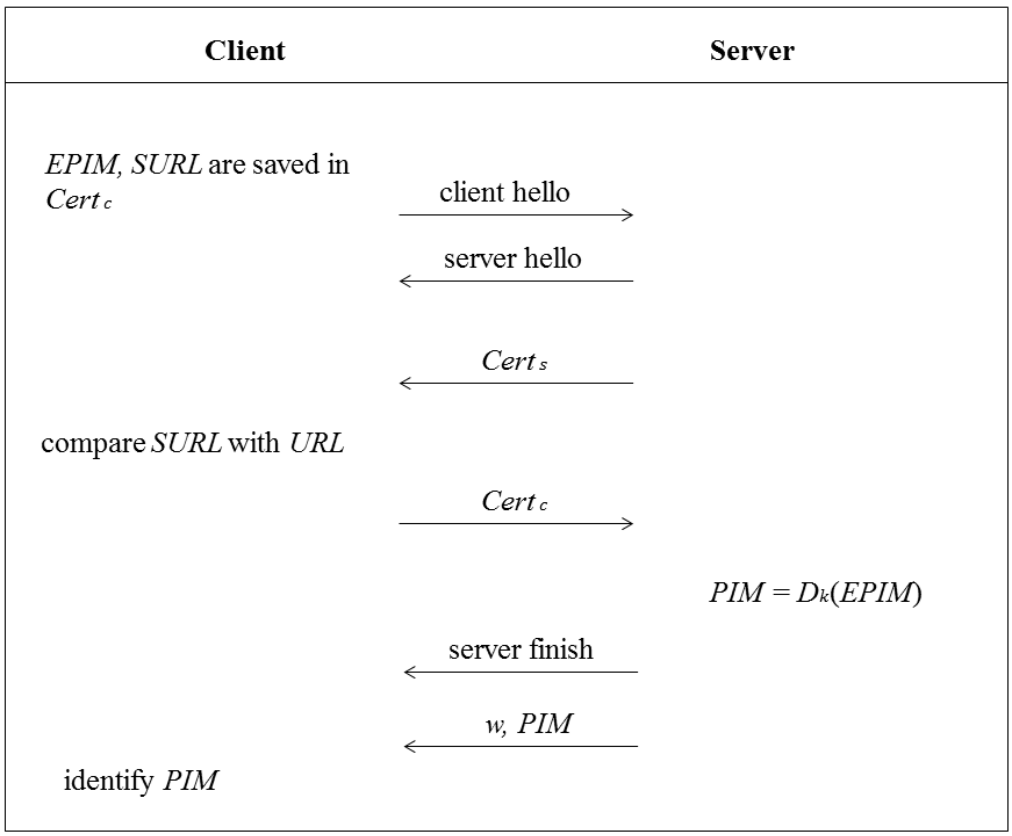

Fig.10 Advanced-SAPIM

5.2.2 Server authentication

Fig.10 shows the server authentication process in Advanced-SAPIM, which proceeds as follows:

1. The server and the customer begin mutual authentication.
2. As the customer receives $Cert_s$, the customer's client compares *URL* with *SURL*. If they are identical, the client sends $Cert_c$. Otherwise, the client does not send it.
3. If the server receives $Cert_c$ from the customer, it decrypts *EPIM* using *k*. Otherwise, the server aborts this protocol.
4. A SSL/TLS session is established successfully.
5. The server sends *w* and *PIM* which is decrypted.
6. The customer receives the *w* and recognizes *PIM*.

6 Analysis

6.1 Preventing attacks

6.1.1 Preventing phishing attacks

SAPIM: SAPIM uses *PIM,* a value chosen by a customer. Thus, a phishing attack can be prevented through the customer's recognition of *PIM.*

Advanced-SAPIM: Since Advanced-SAPIM is based on SAPIM, it also prevents phishing attacks. Advanced-SAPIM prevents a phishing attack using URL Obfuscation because the phishing URL is not identical with *SURL,* which is saved in $Cert_c$.

6.1.2 Preventing active phishing attacks

While SAPIM is not completely secure against an active phishing attack, Advanced-SAPIM can certainly prevent one. A detailed analysis follows.

SAPIM: An active phishing attack under SAPIM is described as follows:

1. An Internet banking customer connects to an attacker's proxy server instead of a genuine banking server. At the same time, the proxy server connects to the actual banking server masquerading as the customer.
2. If the customer sends $Cert_c$ to the malicious server, the attacker relays the $Cert_c$ to the bank server as though transmitted directly from the customer.
3. The bank server sends *w* and *PIM*, which the proxy server relays to the customer.
4. After the customer recognizes her PIM, she enters her ID and password into *w*.
5. The proxy server receives the customer's ID and password, using which the attacker successfully logs into the customer's bank account.
6. SAPIM has failed against an active phishing attack.The attacker will act like the real customer using the ID/PW. Also, the attacker will act simultaneously like the real server.

Advanced-SAPIM: Under Advanced-SAPIM, the response to an active phishing attack is as follows:

1. The customer connects to the attacker's proxy server instead of the genuine bank server. At the same time, the proxy server connects to the bank server pretending to be the customer.
2. When the customer receives $Certs_s$, sent from the bank server and relayed by the attacker, the customer's web client compares *URL* with *SURL,* which is saved in $Cert_c$.
3. Since the URLs are not same, the client does not send $Cert_c$.
4. The bank server does not receive $Cert_c$. Thus, the bank server disconnects a connection with the client of the customer.

6.2 Scheme analysis

In this section, we analyze our proposed schemes using the aforementioned criteria -- usability, deployability and security. Table 1 shows the comparison between our schemes and currently deployed techniques .

6.2.1 Usability

One of the demands of usability is that the bank's server require minimal input from the customer. Bank of America's SiteKey verifies the *PIM* after the customer enters her ID. This is inconvenient, since the customer has to enter her ID every time she wants to connect to the bank's server. Furthermore, in this model, if the customer mistakenly connects to a phishing site, her login ID will become known to the attacker.

However, SAPIM and Advanced-SAPIM do not require any input value from the customer. Furthermore, reliance on the customer's ability to recognize server authentication, as in EV SSL and User-Selected -Image, is also be uncomfortable and dangerous. For example, EV SSL and user-selected-image are used for server authentication. However, if a customer cannot recognize change of address bar or the private image, the customer cannot authenticate a server. SAPIM shares in this problem with these techniques, since its own authentication method involves a PIM. Therefore, in order to be less dependent on PIM for server authentication, Advanced-SAPIM uses the URL comparison method.

6.2.2 Deployability

Our proposed schemes are based on SSL/TLS and so can be integrated into currently used online financial systems. SAPIM in particular is directly applicable without modifying SSL/TLS. However, the application of Advanced-SAPIM to current systems requires some modification, since we have adjusted the SSL/TLS protocol in this scheme to prevent active phishing attacks.

6.2.3 Security

Our schemes are based on SSL/TLS. Thus, our protocols also provide the same security functions as SSL/TLS. Furthermore, as described in section 6.1, SAPIM can effectively prevent phishing attacks and Advanced-SAPIM can effectively prevent active phishing attacks. Thus, Advanced-SAPIM is more secure than currently used server authentication schemes -- EV SSL, User-Selected-Image, PPSBL. Advanced-SAPIM is also more likely to neutralize new phishing techniques, since it uses URL comparison.

Table 1: Scheme analysis table

Criteria	Requirements	EV SSL	User-Selected-Image	PPSBL	SAPIM	Advanced-SAPIM
Usability	Memorywise-Effortless	O	X	O	X	X
	Scalable-for-Users	O	O	O	O	O
	Nothing-to-Carry	O	O	O	O	O
	Physically-Effortless	O	O	O	O	O
	Easy-to-Learn	Δ	O	O	O	O
	Efficient-to-Use	N/A	N/A	N/A	N/A	N/A
	Infrequent-Errors	N/A	N/A	N/A	N/A	N/A
	Easy-Recovery-from-Loss	N/A	N/A	N/A	N/A	N/A
	Non-Recognition	X	X	Δ	X	O
	Minimal-Input	O	X	O	O	O
Deployability	Accessible	N/A	N/A	N/A	N/A	N/A
	Negligible-Cost-per-User	O	O	O	O	O
	Server-Compatible	O	O	O	O	X
	Browser-Compatible	O	O	O	O	X
	Mature	O	O	O	O	O
	Non-Proprietary	N/A	N/A	N/A	N/A	N/A
Security	Resilient-to-Phishing	Δ	Δ	Δ	O	O
	Resilient-to-Active-Phishing	X	X	X	X	O
	Resilient-to-new-type-of-Phishing	Δ	Δ	Δ	Δ	O
	Protocol-Security	O	X	N/A	O	O

7 Conclusion

Phishing attacks pose a serious and costly threat to Internet banking. This threat is compounded by the fact that active phishing attacks are difficult to identify. We have shown that our first scheme, SAPIM, can prevent phishing attacks and makes it more intuitive for the user to identify a connection with a genuine banking server than the SSL/TLS protocol currently in vogue. Moreover, our second scheme, Advanced-SAPIM, can prevent active phishing attacks by using the URL comparison method. Looking to the future, we think that there is a pressing need for efficient server authentication processes that do not rely on the customer's ability to recognize legitimate bank servers in order to eradicate phishing attacks.

ACKNOWLEDGEMENTS
This research was supported by the MSIP(Ministry of Science, ICT & Future Planning), Korea, under the "Employment Contract based Master's Degree Program for Information Security" supervised by the KISA(Korea Internet Security Agency)(H2101-13-1001)

References

[1] S.H. Kim, S.H. Lee and S.H. Jin. 2013. Active Phishing Attack and its Countermeasures, *Electronics and Telecommunications Trends*, Vol 28, No3, pp. 30-50.

[2] Anti-Phishing Working Group. 2012. Global Phishing Survey,
http://docs.apwg.org/reports/APWG_GlobalPhishingSurvey_1H2012.pdf.

[3] Anti-Phishing Working Group. 2013. Global Phishing Survey,
http://docs.apwg.org/reports/APWG_GlobalPhishingSurvey_2H2012.pdf.

[4] Anti-Phishing Working Group. 2013. Global Phishing Survey,
http://docs.apwg.org/reports/APWG_GlobalPhishingSurvey_1H2013.pdf.

[5] CA / Browser Forum. 2012. Guidelines For The Issuance And Management Of Extended Validation Certificates, *www.cabforum.org*, Version 1.4, pp. 1-50.

[6] C. Jackson, D.R. Simon, D.S. Tan and A. Barth. 2007. An Evaluation of Extended Validation Certicates and Picture-in-Picture Phishing Attacks, *Financial Cryptography and Data Security*, Vol 4886, pp. 281-293.

[7] D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley and W. Polk. 2008. Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, RFC5280.

[8] F. Schneider et al. 2009. Phishing Protection: Design Documentation, *Mozila Wiki.*

[9] GlobalSign, *https://www.globalsign.com/ssl-information-.center/what-is-an-ssl-certificate.html.*

[10] Google Safe Browsing, *http:///www.google.com/chrome/intl/ko/more/security.html.*

[11] G. Ollmann. 2005. The Phishing Guide, *Next Generation Security Software Ltd*.

[12] I. Fette, N. Sadeh and A. Tomasic. 2007. Learning to Detect Phishing Emails, *Proceedings of the 16th international conference on World Wide Web*, pp.649-656.

[13] Microsoft Internet Explorer Phishing Filter, *http://windows.microsoft.com/ko* - KR/windows-vista/Phishing-Filter- frequently-asked-*questions.*

[14] J. Bonneau, C. Herley, Paul C. van Oorschot and F. Stajano. 2012. the quest to replace passwords a framework for comparative evaluation of web authentication schemes, *In Proceedings of the 2012 IEEE Symposium on Security and Privacy*, pp. 553-567.

[15] M. Marlinspike. 2009. New Tricks For Defeating SSL In Practice, *Blackhat*.

[16] R. Raines, M. Grimaila, R. Baldwin, R. Bennington and C. 2007. The Use of Attack and Protection Trees to Analyze Security for an Online Banking System, *Proceedings of the 40th Annual Hawaii International Conference on System Sciences*, pp. 144b.

[17] R. Dhamija, J.D. Tygar and M. Hearst. 2006. Why Phishing Works, *In Proceedings of the 2006 conference on Human Factors in Computing Systems(CHI)*, pp. 581-590.

[18] Bank of America SiteKey, *https://www.bankofamerica.com/privacy/online-mobile-banking-privacy/SiteKey.go.*

[19] S. Garera, N. Provos, M. Chew and A. D. Rubin. 2007. A Framework for Detection and Measurement of Phishing Attacks, *Proceedings of the 2007 ACM workshop on Recurring malcode*, pp. 1-8.