SECURITY MEASURES OF RANDVUL KEYBOARD

Radha Damodaram, Lecturer, Dept. of BCA,
CMS College of Science & Commerce

Dr.M.L.Valarmathi, Asst. Professor, Dept. of
Computer Science & Engg, GCT

Abstract

Phishing *is a "con trick" by which consumers are sent email purporting to originate from legitimate services like banks or other financial institutions. Phishing can be thought of as the marriage of social engineering and technology. The goal of a phisher is typically to learn information that allows him to access resources belonging to his victims. The most common type of phishing attack aims to obtain account numbers and passwords used for online banking, in order to either steal money from these accounts or use them as "stepping stones" in money laundry schemes. In the latter type of situation, the phisher, who may belong to a criminal organization or a terrorist organization, will transfer money between accounts that he controls (without stealing money from either of them) in order to obscure the actual flow of funds from some payer to some payee. Phishing is therefore not only of concern for potential victims and their financial institutions, but also to society at large[1].*

In hacker's worlds, there is something called 'Key Logger'. The purpose of key logger is to log every key that you type in your keyboard, this includes every single personal information that you have typed in your keyboard while you surf the Net such as log in into your online banking. Once your password has been logged, the hacker can use your information to their benefit[2]. Using Virtual Keyboard which contains randomly generated keys adds another security layer to authenticate yourself to their system. Virtual Keyboard works just like regular keyboard, one thing is you don't type it in your keyboard. Rather, you will be using your mouse to type the password by using virtual keyboard[3].

KEY WORDS : Keylogging, Random Virtual Keyboardm hackersker, Kernal based, Hook based, RC4 & PRGA..

1.INTRODUCTION

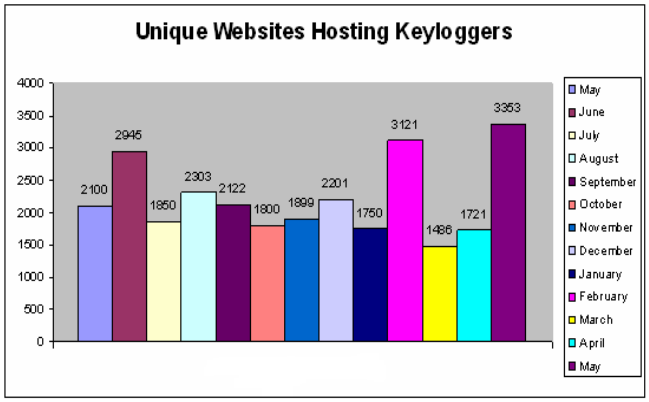
Keyloggers : These are self installing programs and they automatically install themselves either into a web browser or as a device driver, which monitor data being input and send relevant data to a phishing server[4]. Keyloggers use a number of different technologies, and may be implemented in many ways, including:

• A browser helper object that detects changes to the URL and logs information when a URL is affiliated with a designated credential collection site.

• A device driver that stores keyboard and mouse inputs in conjunction with monitoring the user's activities and

• Keyloggers may collect credentials for a wide variety of sites[5]. As with many crime ware varieties, configurators are available to automate construction of customized keyloggers. Keyloggers are often packaged to monitor the user's location, and to transmit only credentials associated with particular sites back to the attacker. Often, hundreds of such sites are targeted, including financial institutions, information portals, and corporate VPNs. Various secondary damage can be caused after a keylogger compromise. In one real world example, a credit reporting agency was targeted by a keylogger spread via pornography spam. This led to the compromise of over 50 accounts with access to the agency, which in turn were used to compromise as many as 310,000 sets of personal information from the credit reporting agency's database[6].

The British Hi-Tech Crime Unit foiled what would have been one of the biggest computer crimes in history, where fraudsters attempted to transfer $420 million from a London branch of Japanese bank Sumitomo Mitsui. The thieves were believed to have hacked into the bank's computer systems using information gathered from keylogger programs, which allowed them access to sensitive passwords and other account information [7].



MAY 2008 TO MAY 2009

Fig.1

The number of unique websites hosting keyloggers hit an all time high in May, 2009 at 3,353. More than 90 percent of spam messages now use HTML to present message content. More than 60 percent of spam messages are sent directly to the recipient's mail server – without passing through any intermediary relay agents.92.99 percent of spam messages are written in English, with German being the next most popular language.

South Korea accounts for the highest source of phishing e-mails–16.33 percent. More than half (55.78 percent) of the world's phishing attacks have fake Web sites hosted in the U.S.

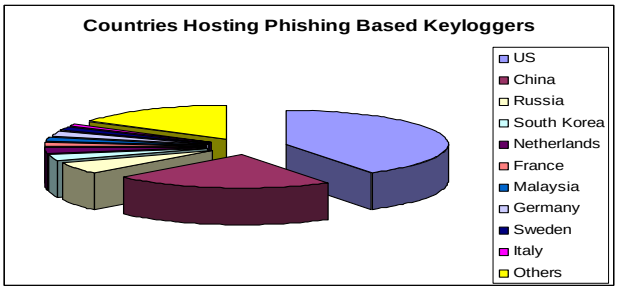


Fig.2

U.S. based businesses are the most targeted organizations of phishing e-mails, accounting for 71.37 percent of all phishing e-mail.Image-based spam has increased linearly since 2005, and accounted for more than 40 percent of spam messages by the end of 2008. The U.S., Spain and France are the three largest originators of spam worldwide.

Countries	Percentage
US	41%
China	22%
Russia	6%
South Korea	3%
Netherlands	3%
France	2%
Malaysia	2%
Germany	2%
Sweden	2%
Italy	1%
Others	16%

Table.1

USA and China each host over 1/3rd of the world's destination websites sent in spam messages.

1.1 DRAWBACKS OF EXISTING SYSTEM

Key loggers are key stroke recorders which can record all keystrokes made on your keyboard! There are two types of key loggers, hardware & software. Hardware key loggers can be attached to keyboard, placed below keyboard or even some key loggers come built in with keyboard.

External key loggers are noticeable by the user, while built in key loggers are not. There are various types of software key [2] loggers. also, kernel based & hook based are popular. Kernel based key loggers resides at kernel level, which gains unauthorized access to hardware (keyboard). Kernel based key loggers can be in the form of keyboard drivers, so be careful while installing drivers from unknown sources.

2 TYPES OF KEYLOGGERS

Keyloggers can be mainly grouped in to 2 major are:
1) Hardware keyloggers, and
2) Software keyloggers.

2.1 Hardware Keyloggers

Hardware keyloggers depend on the hardware components. They are not used in phishing attacks, since these keyloggers require the physical presence to install and to retrieve data from these. They cannot be installed on a largescale also. They lack the anonymity that phishers expect.

2.2 Software Keyloggers

Software key loggers are mostly written for monitoring purposes. Even though they are sometimes put into good use, they usually connote a negative activity.

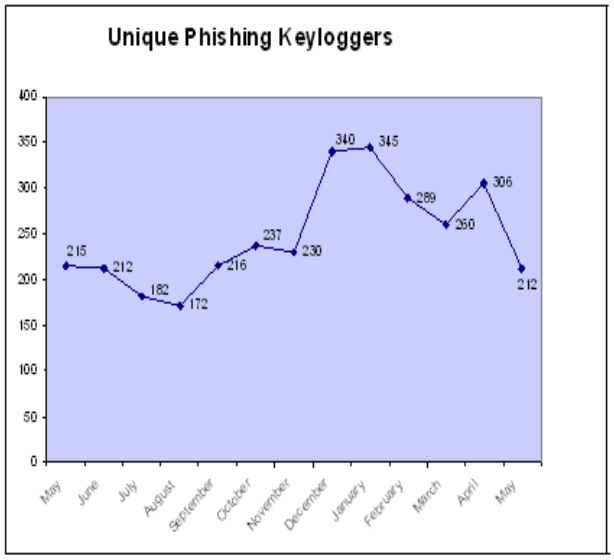


Fig.3

They are the most important weapons in the arsenal of phishers. There are many different variants of keyloggers available now. As many as 212 unique phishing keyloggers were reported by APWG in May, 2009. Because of the ease in which they can be written using any low level language and because of their availaibity either as freeware or shareware, they are mostly employed by phishers.

The key characteristics of a software key logger are:

- Consumption of limited memory resources
- Ability to capture all the available keys on a keyboard
- Ability to create encrypted log files
- Monitoring application usage
- Facility to capture desktop activity and screenshots.
- Facility for sending the log files through email, FTP, Network.
- Ability to disable Anti Keyloggers.

When the security of a system is compromised there is very likelihood of finding Trojans in that system. Almost all Trojans contain code that captures the keys used by the unsuspecting victim. Because of their low resources usage they are not normally traced. Key loggers are mostly used by phisher for identity theft since they can send them all the captured details in a mode defined by them.

3 ANALYSIS OF A KEYLOGGER

A keylogger can either acts as an executable or as a device driver that replaces the existing I/O driver with embedded key logging functionality. Most of the keyloggers written for the Windows operating system work on the principle of hooks. Every key logger consists of a dynamic linking library which contains different functions and an executable file for loading the dynamic linking library and setting the hook. So the successful execution of a keylogger depends on these two files. An understanding of the inner working of a keylogger goes a long way in writing an anti-keylogger.

4. PHISHING BY HOOKS

For keyloggers operating on a windows system, the Windows hook mechanism is the lifeline. A hook is a point in the system message-handling mechanism where an application can install a procedure to intercept message traffic before it reaches a target Window procedure. A function can intercept events before they reach an application through this mechanism.

The function can act on events, modify or discard them. Functions which receive the events are called Filter Functions; every Filter Function is classified by its type. Hooks provide powerful capabilities: Process or modify every message; Record or play back keyboard and mouse events; Prevent another filter from being called; and many more capabilities. Generally, there are two types of hooks: System-wide, and Thread-specific. The System-wide hook is used for filtering messages of all applications (IE: when writing a key logger). And the Thread-specific hook is used for filtering messages of a specific thread. System-wide keyboard hook is used to develop key loggers. To set a System-wide hook we need a DLL.

The reason we need a dynamic linking library for a System-wide hook is because we want the Filter Function to be in any application address space. So when you set the hook message filter function which lies in the .dll , Windows maps the .dll automatically into all applications' address space. Thus we get our filter function called for every process. Therefore when we dynamically link the hook which is in a .DLL it becomes a System-wide hook (of course it depends on the type of Filter Function too). A hook procedure has the following prototype. A hook chain is a list of pointers to hook procedures. When a message occurs that is associated with a particular type of hook, the system passes the message to each hook procedure referenced in the hook chain, one after the other.

A hook procedure can monitor or modify a message passing through a hook chain. It can also prevent the message from reaching the next hook procedure or the target window procedure. The SetWindowsHookEx function installs an application-defined hook procedure at the beginning of the hook chain.

5. RANDOM VIRTUAL (RANDVUL) KEYBOARD DESCRIPTION

When User giving the username and password by use of normal keyboard there will be the possibility of tracing or hacking the username and password. This will be unsafe in Banking system, distributed system and important website. To avoid this type of problem and to give more security to the user we use Randomized virtual keyboard[6].

Randvul keyboard is the dynamic keyboard. The format of keyboard will be changed depends upon the time and client. So each client will have separate keyboard format. When user trying to entering the password and username through normal keyboard, the software will not allow to enter the text in that fields. User must enter the username and password through Randomized virtual keyboard.

After entering username and password this will be encrypted based on presented algorithm. So no one can capture any procedure. This value will be passed as primary key through network. This primary key will be decrypted based on our own algorithm in reverse process method.

So whenever the users open the keyboard the keys automatically swapped from one place to another.

5.1 HOW IT WORKS

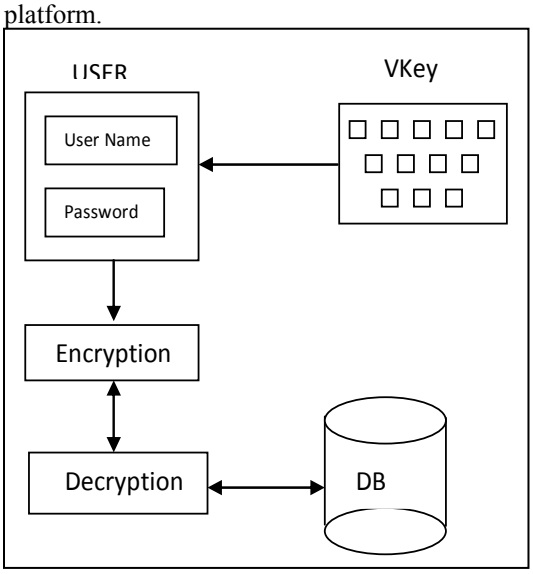
So every time the users got different keyboards. The key entered by users are not directly transferred to online. The keys are encrypted in client side using some user owned algorithm. The reason for using own algorithm is, If we use general algorithms, it's file will then be uploaded to the attacker's website which he can use to his benefit.

These key loggers are easily and freely available on internet and also integrated with other programs like rookits making its detection is very difficult. So your chance of being infected by such a malicious code even if you have an updated antivirus/anti-Spyware program is about 70%. See the brighter side, you still are 30% safe. Sounds scary..huh!!!

So lets keep our focus to internet banking websites and see how we BANK + USERS) can try to avoid compromising login credentials may be chance to identify by hackers. Using encryption algorithm, the key values entered are encrypted and stored in database in unrecognizable format. When we need to login again using the keyboard, the key values are decrypted and the works are reversed.

5.2 . TECHNICAL STRUCTURE

The description of this process is , User enter the username and password using virtual key and that values are encrypted and stored in database. To explain this methodology, there is a design sample secured randomized virtual keyboard in java platform.



5.3 MODULES

5.3.1 Front view design of virtual keyboard

Front view design of virtual keyboard. We can design this module either Microsoft Tech or Sun Micro System Tech.

5.3.2 Random keyboard generation

Random keyboard generation. This module will give random keyboard for every user. Alphabets order will be differ for every user. So that we give more security in distributed system and web application.

5.3 .3 ENCRYPTION & DECRYPTION

Username and password will be given new form in this module. Our own algorithm will follow conversion of the username and password. After encryption that value will be passed to Database server. In this module encrypted value will be decrypted by same algorithm that has used in encryption.

5.4 ALGORITHM DESCRIPTION

Sample Algorithm method used for encrypt and decrypt the key values is RC4 (Ron's Code 4) method. RC4 is a stream cipher symmetric key algorithm.

It was developed in 1987 by Ronald Rivest and kept as a trade secret by RSA Data Security. RC4 uses a variable length key from 1 to 256 bytes to initialize a 256-byte state table. The state table is used for subsequent generation of pseudo-random bytes and then to generate a pseudo-random stream which is

XOR-ed with the plaintext to give the cipher text. Each element in the state table is swapped at least once. The RC4 key is often limited to 40 bits, because of export restrictions but it is sometimes used as a 128 bit key. It has the capability of using keys between 1 and 2048 bits. RC4 is used in many commercial software key. There are two counters i, and j, both initialized to 0 used in the algorithm The random algorithm is used in this keyboard to generate random values each time. The design of this keyboard using eclipse tools. The keyboard buttons are designed Using java swing. Random() keyword is used to generate random numbers between 1 to 36. Because I use 36 keys includes a-z, 0 - 9 for giving passwords[8].

The password field protected from normal keyboard entries. So users must use the randomized virtual keyboards. Packages such as Lotus Notes and Oracle Secure SQL. It is also part of the Cellular Specification.

The RC4 algorithm works in two phases[17]:

1. key setup
2. Ciphering.

1) Key setup

Key setup is the first and most difficult phase of this algorithm. During a N-bit key setup (N being your key length), the encryption key is used to generate an encrypting variable using two arrays, state and key, and N-number of mixing operations. These mixing operations consist of swapping bytes, modulo operations, and other formulae[18].

In the attached project you can see how I do it in the EncryptionKey set property of RC4Engine class.

2) Ciphering phase

Once the encrypting variable is produced from the key setup, it enters the ciphering phase, where it is XOR-ed with the plain text message to create an encrypted message. XOR is the logical operation of comparing two binary bits.

If the bits are different, the result is 1. If the bits are the same, the result is 0. Once the receiver gets the encrypted message, he decrypts it by XOR-ing the encrypted message with the same encrypting variable.

A permutation of all 256 possible bytes (denoted "S" below). Two 8-bit index-pointers (denoted "i" and "j"). The permutation is initialized with a variable length key, typically between 40 and 256 bits, using the key-scheduling algorithm (KSA).

```
 for i from 0 to 255
 S[i] := i
        Endfor
```

Once this has been completed, the stream of bits is generated using the pseudo-random generation algorithm (PRGA). The key-scheduling algorithm is used to initialize the permutation in the array "S". "keylength" is defined as the number of bytes in the key and can be in the range $1 \leq \text{keylength} \leq 256$, typically between 5 and 16, corresponding to a key length of 40 – 128 bits.

First, the array "S" is initialized to the identity permutation. S is then processed for 256 iterations in a similar way to the main PRGA algorithm, but also mixes in bytes of the key at the same time.

```
 j := 0
for i from 0 to 255
 j := (j + S[i] + key[i mod keylength]) mod 256
swap(&S[i],&S[j])
Endfor
```

The pseudo-random generation algorithm (PRGA) which is used to modify the key length of particular key and randomly generating encryption algorithm and is only decrypted by the reverse algorithm of same.

In the attached project you can see how I do it in the RC4 Engine class:

- Encrypt: encrypt method
- Decrypt: decrypt method

In the algorithm the key stream is completely independent of the plaintext used. An 8 * 8 S-Box (S0 S255), where each of the entries is a permutation of the numbers 0 to 255, and the permutation is a function of the variable length

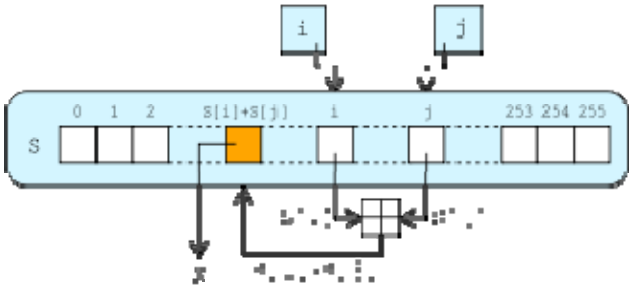


Fig. 5

This Fig. 5 displays how PRGA & RC4 algorithms have implemented. This is the sample only, But the users use any of their own method to give high protections[36].

6. FEATURES

This random Virtual Keyboard adds another security layer to authenticate the system.

- The hacker not able to find which keys are logged by user.
- It can deal with different algorithms by users choice.
- Reverse typing option will helps to provide more security.
- Randomly generating excess letters while encryption and using trim option while decryption.
- Centralized Saving Option.

7. IMPLEMENTATION

Overall we found that the Randvul Keyboard Was examined in this study left a lot to be desired. The experiment shows that it performed best use larger more frequently logged-users. As shown in the Table 1, the rate of false positive and false negative are very low while using Randvul keyboard system and it can handle various categories of keylogging methods. As shown in fig.3 data encryption and decryption methods are very effective and they do not allow any keylogging techniques.

8. CONCLUSION

The most common type of phishing attack that is "Keylogging" aims to obtain account numbers and passwords used for online banking, in order to either steal money from these accounts or use them as "stepping stones" in money laundry schemes.

The purpose of key logger is to log every key that you type in your keyboard, this includes every single personal information that you have typed in your keyboard while you surf the Net such as log in into your online banking.

Virtual Keyboard works just like regular keyboard, one thing is you don't type it in your keyboard. Rather, you will be using your mouse to type the password by using virtual keyboard.

This is when password stealers were simple key loggers. Whatever keys were typed in, they were captured by the malicious code and logged into some file in simple or encrypted form. This file was later uploaded to attacker's site or simply the logs were mailed. This attack can be mitigated by virtual keyboards.

This system works simply by clicking the Randomized keyboard layout by a mouse. So the basic key loggers cannot capture the mouse clicks and hence the passwords cannot be logged.

9. FUTURE WORK

The purpose of key logger is to log every key that you type in your keyboard, this includes every single personal information that you have typed in your keyboard while you surf the Net such as log in into our online banking. In future, The randvul keyboard deals with the following options :

1. Different languages will be implementing by using different algorithms.
2. Security enrichment via Voice Recognizing

References:

[1] APWG. Phishing Activity Trends - Report for the Month of December, 2009. Technical report, Anti Phishing Working Group, May,. 2008. Available at http://www.antiphishing.org/reports/apwg_

[2] Tom Olzak, MBA, CISSP, MCSE, President and CEO of Erudio Security, LLC.

[3] tom.olzak@erudiosecurity.com.

[4] Tom's book, Just Enough Security

[5] CipherTrust Inc., *The Next-Generation Reputation System*, 2005

[6] C. Dwork, A. Goldberg, and M. Naor, *On Memory-Bound Functions for Fighting Spam*, 2004

[7] R. Dhamija and J. D. Tygar. The battle against

[8] C. Dwork and M. Naor, *Pricing via Processing or Combating Junk Mail*, 1992

[9] A. Back, *Hashcash- A Denial of Service Counter-Measure*, 2002

[10] Radicati Group Inc., *Market Numbers Quarterly Update*, Q4, 2003

[11] Brightmail Inc., *Spam Percentages and Spam Categories*, 2004

[12] Internet Systems Consortium, *Internet Domain Survey*, 2004

[13] V. V. Prakash and A O'Donnell, Cloudmark, *Fighting Spam with Reputation Systems*, 2005

[14] A. Oram, ed. *Peer-to-Peer Harnessing the Power of Disruptive Technologies*, Chapter 16, O'Reilly and Associates, Cambridge, MA, 2001

[15] Phishing: Dynamic security skins. In Proceedings of the 2005 symposium on Usable privacy and security, New York, NY, pages 77–88. ACM Press, 2005B. Laurie and R. Clayton, *"Proof of Work" Proves Not to Work*, 2004

[16] R. Jurca and B. Faltings, *Reputation-based*

[17] Compaine B. J., 1988, *Issues in New Information Technology*, Ablex Publishing; Norwood, NJ.

[18] Computer Science and Telecommunications Board, 1999, *Trust in Cyberspace*, National Academy Press, Washington, D.C.

[19] Dawes, McTavish & Shaklee, 1977, "Behavior,communication, and assumptions about other people's behavior in a commons dilemma situation," *Journal of Personality and Social Psychology*, Vol 35, pp 1-11

[20] Foley, 2000, "Can Micrsoft Squash 63,000 Bugsreport_dec_2007.pdf.

[21] H. Aradhye, G. Myers, and J. Herson. Image analysis for efficient categorization of image-based spam e-mail. Document Analysis and Recognition, 2005.Proceedings. Eighth International Conference on, 2:914–918, 29 Aug.-1 Sept. 2005.

[22] Scott Fluhrer Itsik Mantin and Adi Shamir department The Weizmann Institute Rehovot Israel fitsik shamirg wisdom weizmann ac il rc4_ksaproc.pdf

[23] Lars R. Knudsen1, Willi Meier2, Bart Preneel3?, Vincent Rijmen3, and Sven HTL Brugg-Windisch, CH-5210 Windisch "Analysis Methods for (Alleged) RC4".

[24] N. Chou, R. Ledesma, Y. Teraguchi, D. Boneh, and J. Mitchell. Client-side defense against web-based identity theft. In 11th Annual Network and Distributed System Security Symposium (NDSS '04), San Diego, 2005.

[25] R. Dhamija, J. D. Tygar, and M. Hearst. Why Phishing Works. In Proceedings of the Conference on Human Factors In Computing Systems (CHI) 2006, Montreal, Canada. ACM Press, 2006.

[26] E. Kirda and C. Kruegel. Protecting Users Against Phishing Attacks with AntiPhish. In COMPSAC '05: Proceedings of the 29th Annual International Computer Software and Applications Conference (COMPSAC'05) Volume 1, pages 517–524, Washington, DC, USA, 2005. IEEE Computer Society.

[27] E. Kirda and C. Kruegel. Protecting Users against Phishing Attacks. The Computer Journal, 2006.

[28] E. Medvet, E. Kirda, and C. Kruegel.Visual-Similarity-Based Phishing Detection. Technical Report, TR-iSecLab-0708-001, http://iseclab.org/ papers/visual-phishing-technical.pdf, 2008.

[29] Microsoft. Sender ID Home Page. http://www.microsoft.com/mscorp/safety/technologies/senderid/default.mspx, 2008.

[30] P. Mutton. Italian Bank's XSS Opportunity Seized by Fraudsters. Technical report, Netcraft, Jan. 2008. Available at http://news.netcraft.com/archives/008/01/08/italian_banks_xss_opportunity_seized_by_fraudsters.html.

[31] NetCraft. Netcraft anti-phishing tool bar.http://toolbar.netcraft.com, 2007.

[32] A. Rosiello, E. Kirda, C. Kruegel, and F. Ferrandi. A Layout-Similarity-Based Approach for Detecting Phishing Pages. In IEEE International Conference on Security and Privacy in Communication Networks (SecureComm), 2007.

[33] B. Ross, C. Jackson, N. Miyake, D. Boneh, and J. C. Mitchell. A Browser Plug-In Solution to the Unique Password Problem.

[34] http://crypto.stanford.edu/PwdHash/, 2005.

[35] B. Ross, C. Jackson, N. Miyake, D. Boneh, and J. C. Mitchell. Stronger Password Authentication Using Browser Extensions. In 14th Usenix Security Symposium, 2005.

[36] F. Schneider, N. Provos, R. Moll, M. Chew, and B. Rakowski. Phishing Protection Design Documentation. http://wiki.mozilla.org/Phishing_Protection: Design_Documentation, 2007.

[37] Sophos. Do-it-yourself phishing kits found on the internet, reveals Sophos. Technical report, Sophos, Aug. 2004. Available at http://www.sophos.com/pressoffice/news/articles/2004/08/sa_diyphishing.html.

[38] SpoofGuard. Client-side defense against web-based identity theft. http://crypto.stanford.edu/SpoofGuard/, 2005.

[39] R. Stankovic and B. Falkowski. The Haar wavelet transform: its status and achievements. Computers and Electrical Engineering, 29:25–44, 2003.

[40] Z. Wang, W. Josephson, Q. Lv, M. Charikar, and K. Li. Filtering Image Spam with Near-Duplicate Detection. Proceedings of CEAS 2007: Fourth Conference on Email and Anti-Spam, Aug., 2007.

[41] L. Wenyin, G. Huang, L. Xiaoyue, Z. Min, and X. Deng. Detection of phishing webpages based on visual similarity. In 14th International Conference on World Wide Web (WWW): Special Interest Tracks and Posters, 2005.

[42] Yahoo. Yahoo! AntiSpam Resource Center. http://antispam.yahoo.com/domainkeys, 2008.

[43] B. Schneier, Applied Cryptography, Wiley, New York, 1996.

[44] J. Dj. Goli¶c, \Linear Statistical Weakness of Alleged RC4 Keystream Generator," Advances in Cryptology - EUROCRYPT'97, Lecture Notes in Computer Science, Vol. 1233, Walter Fumy (Ed.), Springer-Verlag, pp. 226-238.

[45] R. A. Rueppel, Analysis and Design of Stream Ciphers, Springer-Verlag, Berlin, 1986.