WHITE PAPER: SYMANTEC SECURITY RESPONSE

# Threats to Online Banking

Candid Wüeest
Symantec Security Response, Dublin

# Threats to Online Banking

## Contents

## Abstract
The number of malicious applications targeting online banking transactions has increased dramatically in recent years. This represents a challenge not only to the customers who use such facilities, but also to the institutions who offer them, as evidenced by an ongoing trail in the US. These malicious applications employ two kinds of attack vector – local attacks which occur on the local computer, and remote attacks, which redirect the victim to a remote site. The possibility also exists that both approaches will be combined. Some attacks may be foiled by adopting security measures such as transaction numbers (TAN). However, it is likely that the risks associated with online banking transactions will remain until new transaction methods, such as PKI based methods (public key infrastructure), are widely introduced.

## Introduction
A Miami businessman is suing his bank for the loss of $90,000. He claims that, in February 2005, this money was stolen from his online bank account via an unauthorized transaction. Investigations have revealed that the businessman's computer was infected with a Trojan capable of logging keystrokes, including his full account details. It is likely that the theft of this information was the trigger that led to the unapproved transaction to a foreign bank account. So far, the businessman's bank has refused to compensate for his loss [1].

This fraud case is not an isolated incident. The prevalence of malicious applications that steal financial account information has increased dramatically over the last year, often resulting in victims losing hard currency. In May 2003, only around 20 such Trojans existed in the wild. Two and a half years later, the number has increased to more than 2,000 [2]. Why?

There are several factors that may have influenced the evolution of this type of malicious application, but maybe the dramatic increase in their prevalence is just because they have a higher chance to succeed than expected. The case of the businessman in Miami is just one example of many that have succeeded.

## Evolution
In the early stages, Trojans that steal financial account information targeted only a handful of online banks. For example, in August 2003, PWSteal.Bancos.B [3] stole account information from only five banks.

This has changed dramatically. PWSteal.Bancos.T [4], discovered in April 2005, contains a list of 2,764 URLs from 59 different top-level domains. The corresponding organizations range from small local bank branches to international banking groups. In February 2005, a Trojan named Trojan.Goldun.B [5] was discovered stealing account information for an online payment service called e-gold. The Trojan disguised itself as a security update for e-gold. When a user executed the deceptively-named file SecurityEgold.exe', the Trojan registered itself as a browser helper object (BHO) and monitored Internet Explorer for visits to pre-defined URLs. Any account information that was gathered by the Trojan was posted via a PHP script on a domain controlled by the attacker. The log file on the server storing the data was accessible by

anyone, most likely due to a misconfigured web server. A quick look at this log file showed a growing list of account numbers and corresponding passwords. Within an hour, the PHP script had added another 13 valid-looking account credentials. The site was online for another 24 hours before being shut down.

Interpolation of this data leads to the conclusion that the Trojan.Goldun.B attacker had received details for a large number of accounts, providing him with the opportunity to steal hard currency from the victims. The attack vectors used by this kind of malicious application can be categorized in two groups: local and remote attacks. Local attacks happen on the local computer during an online banking session. Remote attacks do not execute code on the local computer, but redirect the victim to a remote site.

## Local Attacks

A common mistake made by end users is believing that their online banking session is perfectly safe when they use an SSL connection. Security experts continually state that everything is safe if there is a yellow padlock symbol in the browser window.

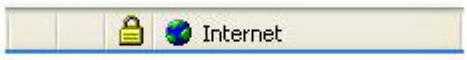


**Figure 1: The yellow padlock symbol as displayed in Internet Explorer.**

But SSL is designed as a secure tunnel from the end user computer to the bank mainframe and does not protect the end points such as the end user's computer. The PWSteal.Bankash.A [6] Trojan exploits this fact. The Trojan drops a DLL and registers its CLSID as a browser helper object in the registry. Thus the Trojan is able to intercept any information that is entered into a web page before it is encrypted by SSL and sent out. This functionality can also be achieved by injecting the Trojan directly into the web browser's memory space, which also can often bypass desktop firewalls when making outgoing connections. Other local attack methods include running a layered service provider (LSP) monitoring all network traffic, writing its own network driver, or displaying a carefully crafted copy of a website on top of the official website.

The PWSteal.Bancos family does the latter. When an infected user visits one of the predefined domain names, the Trojan generates a pop-up window which overlays the current browser window. The pop-up window contains an exact copy of the original service website login page. When information is entered into the fake form and the send button is pressed, the spoofed pop-up window closes, leaving the old browser window. Meanwhile, the harvested account credentials are sent to a remote server. These are only some of the possible methods that will work even if the session is SSL encrypted. These procedures will also bypass the virtual keyboard – a countermeasure that has been introduced by some online banking systems against key loggers. Here, the user clicks on a virtual keyboard displayed on the screen, rather than pressing the real keys on their keyboard, but this only shifts the problem: screen captures,

fake website pop-ups and malicious code running inside the web browser can record exactly the same information as key loggers. No matter how the information is entered into the web form, once it is entered, it can be intercepted.

A further step towards better security therefore, would be the use of non-static user credentials. A user name and a static password are simply no longer enough to protect online banking sessions. Some companies have already responded to these threats by introducing dynamic passwords including RSA secured ID tokens [7] or one-time passwords on paper lists called transaction numbers (TAN).

Unfortunately this does not solve the problem entirely. Since the method for entering authentication data has not changed, the password still can be intercepted. The only additional hurdle is that the attacker must use it first, before the legitimate user does. This behavior has not yet been observed in the wild, but consider the following scenario: a Trojan, intercepting a password by any of the discussed methods, simply has to send this information to the attacker. Meanwhile, it blocks every other connection from that computer. Thus the current online banking session is never completed. A network driver, LSP, BHO, or rootkit that hooks network API calls could do this. All the attacker needs to do is to use the unused, one-time password quickly to establish an online banking session from his or her own computer, enabling the attacker to do whatever he or she likes. Some companies therefore ask for multiple TANs during a session, one for login and one for each transaction made. Others ask for a specific TAN on a list and the position is chosen at random each time. Still you do not achieve 100 per cent security, as a man-in-the-middle attack can trick the user into revealing this information.

One next step towards better security may be PKI (public key infrastructure) smartcards, which have already been introduced by some banks. These cards can be attached to the computer using a USB card reader and can act as a challenge-response authentication or use a zero knowledge authentication, leaving the attacker with little useful information.

### Remote Attacks

Usually, the attacker sets up a copy of the web page he wants to impersonate on a server he controls. In the past attackers often linked directly to the original images on the legitimate web server, which left easy-to-follow traces in the webmaster's log files. Nowadays, attackers tend to keep resources locally. Once the bait server has been set up, the attacker sends out emails that trick the user into visiting the spoofed website. These emails often prompt the user to visit the online service in order to provide some urgent data verification, or indicate that the user is required to visit the website because of some update process in the main database of the service provider. This form of social engineering attack, with the goal of acquiring user account information, is also known as phishing.

The location of the real server is obfuscated or masked using exploits or by other, not so well known methods. For example, one can translate the quartet of an IP address (such as 216.239.57.99) into a decimal number (such as 3639551331). Then a fake user authentication can be added that is made to

look like the impersonated domain (http://mySecureBank.tld@3639551331). This trick fools some users into believing that they are clicking on a link that leads to the mySecureBank.tld domain. Instead, it goes to 3639551331, which is the IP address of google.com represented as a decimal number. The use of international domain names (IDN), introduced by ICANN in June 2003, adds a further complication to the matter of identifying URL obfuscation. The fact that international characters can be used in domain names raises the issue of domain names that have been spoofed simply by replacing some of the letters in their name with letters from different alphabets that look the same.

For example, an attacker could register the domain 'mySecureBank.tld', where the 'a' is replaced with an 'a' from the Cyrillic character set, which looks identical. If a hacker finds a domain-authenticated SSL service, he or she can even add an SSL padlock in his attempt to fool the end user, as demonstrated by Johanson [8]. Trojan.Blinder [9] utilizes another example of obfuscation. Trojan.Blinder uses JavaScript to layer a white box over the location field of the browser hiding the fake URL. The box contains a spoofed URL that looks like a legitimate website. The position of the box is even recalculated and reapplied multiple times per second to ensure a seamless integration with the browser.

Furthermore, obfuscation is not even necessary. As seen in February 2005, large DNS poisoning attacks can lead to browser redirection, without even modifying the end user's computer directly. Once the user is on a spoofed website everything entered there can be captured.

### Joint Forces

If an attacker combines local and remote attacks more serious damage can result. For example, a Trojan running on an infected computer can alter the local hosts file to redirect any requests for mySecureBank.tld to an IP address controlled by the attacker. This behavior has already been observed in a number of adware threats in the wild. To complete the illusion, the Trojan can also install a self-signed root certificate on the infected computer. Free tools like OpenSSL can be used to help create these certificates. This enables the attacker to generate official-looking SSL connections from the infected computer to the malicious web server hosting the spoofed website. The chances of an average user noticing these changes are very slim.

Once the user has been trapped on such a spoofed website, the attacker can act as man-in-the middle and relay any challenge-response protocol that might be implemented by the original online banking system. At the moment we are not aware of a Trojan in the wild performing such an attack, but that does not mean that there couldn't already be one doing this. Such an attack could be countered by carefully checking the IP addresses involved in the session and their owners.

## Conclusion

These examples show that the biggest threat to online banking is still malicious code executed carelessly on the end-user's computer. The attackers tend to target the weakest link. Once the attacker has control over a user's computer, he or she can modify the information flow to his or her advantage. This may have happened in the case of the businessman from Miami.
The situation most likely will not change until new transaction methods are introduced.
So, whenever using an online financial system today, ensure that your system is still under your control and not a spoofed puppet, or you could end up featuring as the businessman in the next fraud case article.

## References

[1] John Leyden, 'Florida man sues bank over $90k wire fraud',
http://www.theregister.co.uk/2005/02/08/e-banking_trojan_lawsuit/.
[2] Joakim von Braun, Symantec, Sweden.
[3] For detailed information on PWSteal.Bancos.B (MCID 3487), see
http://securityresponse.symantec.com/avcenter/venc/data/pwsteal.bancos.b.html.
[4] For detailed information on PWSteal.Bancos.T (MCID 4930),
http://securityresponse.symantec.com/avcenter/venc/data/pwsteal.bancos.t.html.
[5] For detailed information on Trojan.Goldun.B (MCID 4373), see
http://securityresponse.symantec.com/avcenter/venc/data/trojan.goldun.b.html.
[6] For detailed information on PWSteal.Bankash.A (MCID 4326), see
http://securityresponse.symantec.com/avcenter/venc/data/pwsteal.bankash.a.html.
[7] RSA SecurID Authentication, http://www.rsasecurity.com/node.asp?id=1156.
[8] Eric Johanson, 'The state of homograph attacks', http://www.shmoo.com/idn/homograph.txt.
[9] For detailed information on JS.Trojan.Blinder, see
http://securityresponse.symantec.com/avcenter/venc/data/js.trojan.blinder.html.

## About the Author

Candid Wüest graduated in computer science at the Swiss Federal Institute of Technology (ETH). He extended his experience in IT security during the last ten years while he worked for several companies, including the global security analysing laboratory at IBM Research, Rüschlikon. He has been with Symantec for the last two years and is currently working as a virus analyst in Symantec Ltd. Dublin. He has published various papers and articles in magazines, given interviews to radio and newspapers, and presented at many conferences, such as COMDEX Scandinavia.

## About Symantec

Symantec is the global leader in information security, providing a broad range of software, appliances, and services designed to help individuals, small and mid-sized businesses, and large enterprises secure and manage their IT infrastructure. Symantec's Norton™ brand of products is the worldwide leader in consumer security and problem-solving solutions. Headquartered in Cupertino, California, Symantec has operations in 35 countries. More information is available at www.symantec.com.

Symantec has worldwide operations in 35 countries. For specific country offices and contact numbers, please visit our Web site. For product information in the U.S., call toll-free 1 800 745 6054.

Symantec Corporation
World Headquarters
20330 Stevens Creek Boulevard
Cupertino, CA 95014 USA
408 517 8000
800 721 3934
www.symantec.com