Internet Banking

How secure is it?

Authors: Galli – Musa – Turan

Oregon State University
Department of Electrical and Computer Engineering

ECE578 – Spring 2000

Contents

1 Introduction

Internet technology is making sweeping changes in global communications and how business is conducted. Increasingly, these changes are being felt by financial institutions as their customers request financial services that are more convenient (“anytime, anyplace”) and less expensive. These services all require privacy and integrity of transactions, and most importantly strong authentication. However, while security of the communications is usually well considered, most online banking systems do not sufficiently address the security of the platform on which these applications run. It is well known that many popular operating systems do not provide sufficient protection from malicious programs such as viruses. Viruses may be used to subvert authentication protocols by capturing authentication information such as PINs or passwords that are entered by users.

In this report we will illustrate and give examples of two common electronic banking schemes. These schemes concentrate on the security of the communications between the customer and the bank, but do not provide sufficient protection from other programs on the platform that the customer is using. The aim is to demonstrate that such concerns are very real, and that users, and implementers of such schemes should be aware of the risks they are taking when deploying Internet banking solutions.

The rest of this report is let out as follows: Chapter 2 gives an overview of electronic banking schemes and chapter 3 gives detailed examples of Internet banking solutions from two different countries and some of their weaknesses. Chapter 4 describes other solutions and their vulnerabilities. Chapter 5 describes the security of SSL. Chapter 6 outlines conclusions followed by references at the end.

2 Problems with Secure On-line banking

Electronic banking aims to provide easy access to banking services for customers. Both banks and customers stand to benefit from the introduction of electronic banking schemes, since the bank can offer its services at much lower cost, while the customers can access the services from any location at anytime.

There are a number of models that can be adopted to implement such electronic banking systems. They are described below.

Customers can use existing browser software such as Netscape's Navigator or Microsoft's Internet Explorer as the client interface to the banks system. In this model, the bank's server provides HTML forms-based interface through which customers can make requests and conduct transactions. Communications security is provided by the SSL protocol, which is built into the browser.

Customers can download Java applets from the bank-server's web site. The downloaded applet provides the interface through which customer transactions can take place. In this case, communication security is provided by the applet itself. The benefit of this approach is that the applet can be dynamically updated with new features and that the application will be able to be run on a wide variety of platforms.

Customers can download executable files from the bank-server's web site, which have been precompiled for a number of common platforms. The running executable provides the customer interface as well as the communications security.

Services offered via Internet banking schemes are similar to those offered by many phone-banking services and includes facilities that allow the user to:

- Query account and loan details.
- Obtain records of previous transactions.
- Set up and adjust automatic bill payment schemes.
- Transfer funds between accounts.

Most Internet Banking applications authenticate users based on an account/client number and a secret PIN. The account number is generally typed directly into a text field, while the "secret" PIN is entered with the keypad. Account numbers are displayed in plain text, while an "*" is written to the screen as each keypad key is pressed. Once the user's information is complete, the application contacts the bank server and attempts to authenticate the user.

Network security is well addressed in all Internet Banking solution and is based on establishing a secure channel between the user's client application and the server running at the bank. The protocols used provide connection security that has three basic properties:

The transaction is private. Public key encryption is used after an initial handshake to negotiate a session key. Symmetric cryptography is then used for subsequent data encryption.

The peer's identity can be authenticated using public key cryptography.

Message transport includes message integrity check.

Secure communication based on 1024bit RSA and 128 IDEA provide secure channels that are infeasible to break with current technology. Unless a weakness is found in the IDEA algorithm, 128 bit IDEA symmetric keys can only be cracked by brute force, a process which would take a billion processors which can try a billion keys a second, longer than the age of the universe.

In summary, the communications side of the Internet banking applications would seem to be secure, and should remain so for the foreseeable future. However, as we shall see in the sequel, such security can be easily subverted, if the platforms on which these secure applications run do not protect them from other malicious programs.

2.1 The computing base

As discussed in the previous section, the problem of ensuring privacy and authenticity between peers over a public network is well studied, and solutions are widely available to implementers. However, general attention has been strongly focused on the issues of network security, and not focused enough on the issues of security at the user's platform.

MS DOS has never been evaluated under the US National Computer Security Center (NCSC) Trusted Computer Systems Evaluation Criteria (TCSEC). However, according to the NCSC, "Without modification, it is apparent from the most cursory examination that DOS does not implement many of the features required by the C1 class". Some unevaluated implementations of Unix would also fail to achieve a C1 class rating.

Most personal computing platforms are not designed to enable segregation of users and data, and do not have the facilities to enforce user authentication and auditing, mandatory or discretionary access control. An application downloaded or installed by one user can generally be run by a second, sine there is no concept of file or process ownership or protection.

In the case of Windows 95, there is no security kernel that provides access control to resources. Any application can access file on the system, and can observe (perhaps alter) the flow of messages between hardware devices such as mice and keyboards and applications.

Hence we must consider that any secure application we deploy will be operating in an untrusted environment. Therefore the design of such application must take this into account. There are many imaginable scenarios that could be used to illustrate dangers inherent when executing critical applications in such a not-trusted environment. However, we will center on just one scenario in which a user executes an arbitrary piece of code without even realizing they are doing so – the computer virus.

2.1.1 Viruses

A virus is generally a small program that is able to replicate itself when executed and perform some unsolicited action, either benign or malicious. Viruses can take the form of executable programs or system files that are introduced onto computers via downloaded/copied infected programs or corrupted floppy disks. Viruses can also take the form of programs written in an application's macro language. Such macro viruses can be attached to files such as word processing documents intended for the target application, and will be executed when the file is processed.

2.1.2 Relevance to Internet banking

When a virus runs on a platform that has no concept of ownership or access control, any application that runs on that platform can be compromised. Even if access control is enforced by the OS or by additional hardware, the virus techniques can still work. While the OS process management can ensure that other users' processes are safe from direct intervention by the malicious process, the process owned by the owner of the virus can still be compromised.

In the following we describe two different attacks.

2.1.2.1 Attack 1

When users enter their card number using the keyboard, they are forced to enter their PIN with the virtual keypad (prevent key press loggers). As each key on the virtual PIN pad is clicked as an "*" is displayed in an edit control on the screen. Unfortunately for the designers, while this method does stop attackers from logging key presses, the PIN box is a standard edit control, which while displaying "*"s on the screen, also holds the plaintext PIN in its buffer. IT is possible to retrieve this information quite easily using a technique called text scraping.

Windows 95 uses a message-based scheme to control user input. Each window has a message queue that contains messages such as "key pressed", "mouse click" and so on. On receiving each input message, a program will either process it or pass it to the OS to handle. These messages may also include request such as "give me your window ID". In addition to these messages, Windows 95 also provides notification when notable events occur such as "window minimized" or "new application starting".

In this attack, a malicious program can wait for notifications about new windows opening, and wait for the sign of the bank's main application window before activating its malicious code. The malicious program would determine the ID of the OK button, which a user clicks when he has entered his card and PIN correctly. The malicious program then scrapes the authentication info from the form sending the messages in the PIN buffer.

2.1.2.2 Attack 2

In this second attack we assume that the text in the PIN buffer is encrypted with a simple substitution cipher, which varied from execution to execution. This approach simply captures a bitmap image of the keypad window after each mouse click and drawing a red dot corresponding to the co-ordinates of the mouse pointer in that window. The images corresponding to each buttons can then be sent to a server.

3 Examples of Internet Banking Interfaces

We will describe in detail now the Internet Banking solutions of two selected banks . We do not consider those two institutions as especially good or bad examples in terms of security. We chose banks from different countries and different sizes.

3.1 Credit Suisse (CS), Switzerland

This chapter describes the Internet banking solution 'Direct Net' of Credit-Suisse AG, Zurich, Switzerland [CSweb]. The Credit-Suisse Group is the second largest bank in Switzerland and therefore among the largest banks in the world.

3.1.1 Services

Credit Suisse provides an extensive service over the Internet. Its interface allows you to manage several bank accounts, doing worldwide transactions and stock exchange.
They offer two different user interfaces, an HTML-interface that can be accessed over a web-browser and a Java-based finance application that also allows offline transactions.

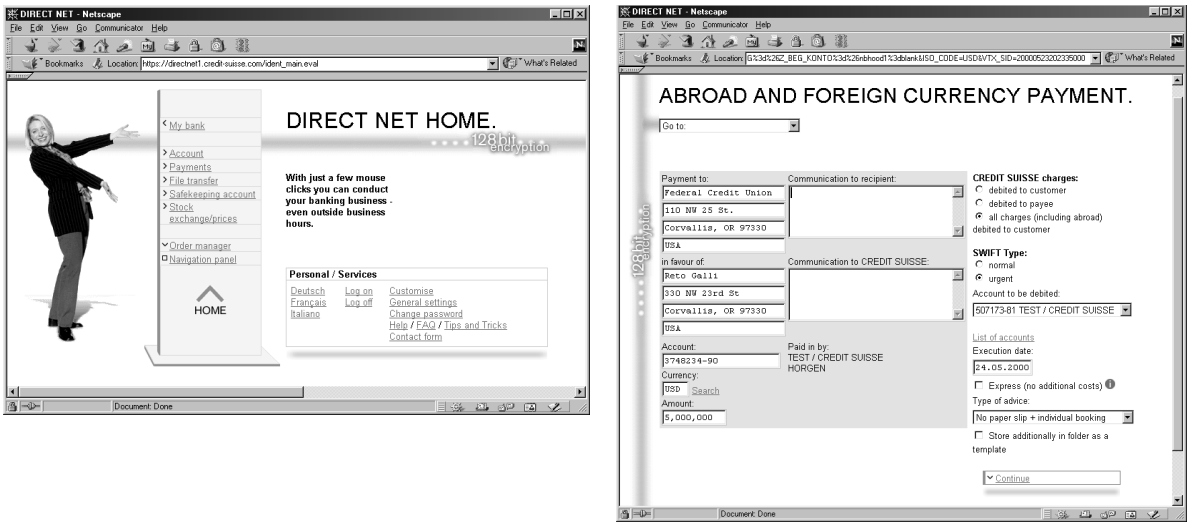

Figure 3-1: Direct Net HTML-Interface

3.1.2 Requirements – Software

To use the HTML interface you need Netscape Communicator 4.5 or higher or Microsoft Internet Explorer 5.0 or higher. You can either have a browser version with 128bit encryption or additional encryption software like SafeWeb. SafeWeb (for Windows or Macintosh) can be downloaded from the CS Homepage.

To use Internet Banking with the Java-based application one needs to download the appropriate package from the Credit Suisse web site. The Java application includes 128bit SSL encryption.

Both interfaces require a connection to the Internet over a LAN or a fast modem or a direct dial to connect to CS. The direct dial to CS in is only possible in Switzerland.

3.1.3 Security Features

The connection from the user's computer to the CS server is secured by SSL with 128 bit encryption and 1024 bit certificates. According to US laws browsers can only change to this encryption mode when the server provides a valid certificate. CS is certified by:

Verisign, Secure Server Certification Authority, RSA Data Security, Inc., US.

3.1.3.1 Authentication

The user has to authenticate itself with a user-number issued by CS, a 7 or 8 character password chosen by the user and a changing access number.

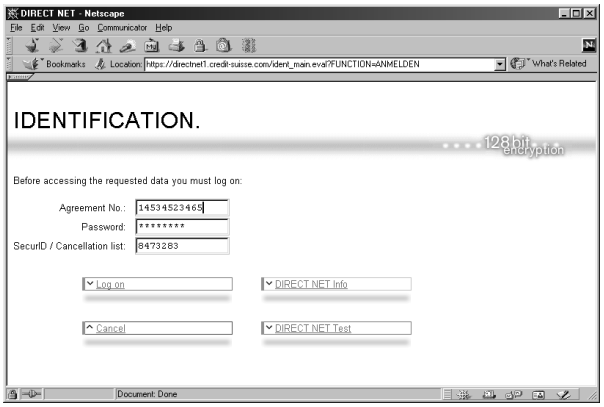

Figure 3-2: Direct Net logon screen

The changing access number comes for infrequent users from a cancellation list or for frequent users from a so-called Secure ID Card. The second is a calculator in credit card format that shows a different random number every minute. The bank sends the authentication information to the customer by regular mail.

The user can freely choose his password. The system prompts the user to select a password that is not easily guessable, but it also accepts input like 'aaaaaa'.

If a user fails to authenticate three times in a row, the system automatically blocks the access to Direct Net. Access can only be obtained with the users written request.

3.1.3.2 Software Download

The download of the Java Internet-Banking application is realized from an encrypted and certified page (SSL). This means the user can be sure that he is downloading the software from a CS server and not from a fake CS download page with a Trojan application.

3.1.4 Information for the users

Information about security issues is unfortunately not well presented and therefore not easy to find. Some general information including good information about the danger of Viruses and Trojans can be found on the information page that advertises their Internet Banking Product. On the same page they also say: "Encryption guarantees security" a false statement from some PR guys that needs no further comments.

More detailed information can be found in the user area after login. Under 'Help' they have a whole section about security. Unfortunately the information about Viruses and Trojans is not to be find here. A user who is already registered for Internet Banking will probably never go to the advertisement page again and therefore never get this important information.

Besides the fact that the user has to gather security information from different places and will maybe never find some important things we found the following weaknesses in the security information:

- The Microsoft Internet Explorer 5.x with standard security settings stores the contents of an encrypted webpage unencrypted in the cache on the local hard-drive. The contents of the browser cache can easily be read from applets or active-X controls from other webpages. Therefore is it important to change the settings of IE 5.x. CS does not mention this essential change of the IE settings in their user information.
- They tell you to check the certificate of the web-server before you login, but they do not tell you how to do that and they provide no information about the valid certificate (Issued by, Fingerprint…)
- Netscape Communicator 4.05 to 4.72 have a vulnerability in checking certificates [sf1188]. CS still recommends using NS 4.50 or higher.

3.1.5 Legal Terms

In this section we summarize some points of the terms and conditions the user has to accept when he uses Direct Net.

3.1.5.1 Liability of the bank

About one third of the terms and conditions are about the non-liability of the bank.

The user has to acknowledge a list of possible risks, including the danger that third parties could gain unnoticed access to a customer's computer during a connection to the bank and could record communication with the bank.

The bank assumes no responsibility for the accuracy of the data transfer to the bank. And furthermore they accept no liability for any software they provide. That means they are no liable if somebody can hack the users bank account due to a weakness in their software.

3.1.5.2 Banking confidentiality

Although individual data packages are transmitted in encrypted form, the sender and recipient are not encrypted and can thus be read by third parties. It is therefore possible for a third party to discover that a banking relationship exists.

3.1.6 System Weaknesses

The Internet-Banking system from CS is very well developed. Some vulnerabilities we found in other solutions are solved in this system.

Of course there are possibilities to attack the system. It is impossible for the bank to guarantee that there are no flaws in the web-browsers that implement SSL and check the certificates. The system can also only be as secure as the computer it is installed on. It will always be possible to attack such a system by Trojan application or Viruses that spoofs passwords and other information. It is up to the user to protect his computer against those attacks.

The information for the users about security issues can be improved. The important information about Trojans and Viruses (possibly the most likely attack) should be at the same place as the other security information and the missing information mentioned in 3.1.4 should be added.

3.2 Credit Mutuel, France

In this paragraph, we will describe the Internet banking solution 'CyberMUT' of Credit Mutuel, France.

3.2.1 Services

Credit Mutuel provides online services to its customers through HTML pages. Their interface allows users to manage their bank accounts, their stocks, their life insurance etc…

3.2.2 Authentication

The authentication of users is based on an account number and a secret password. The account number is directly typed into a text field and printed in clear while the secret password is displayed as "*"s. Once the user information is complete, the application contacts the bank and attempts to authenticate the user.

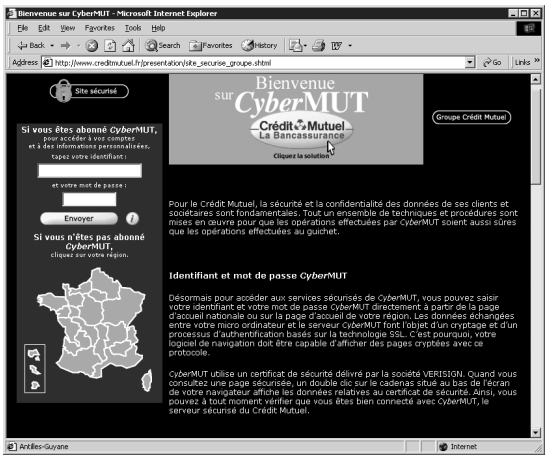

Figure 3-3: The login interface of CyberMUT

3.2.3 Security issues and system weaknesses

A thorough study of CyberMUT brought us to some conclusions with regard to the security of this Internet banking solution.

- The login identifier is the user's account number. This piece of information can be easily retrieved from any mail from the bank to the user. It can also be retrieved at an ATM window where most of the receipts are thrown away.
- The password is selected by the user without size and pattern restriction. A user can select a one-character password, which is accepted by the system.
- The user can navigate in and out CyberMUT without re-entering his password. Hence leaving the browser open after any transaction may be very risky for the overlooking user.
- Due to French regulations, encryption strength is restricted. Hence CyberMUT is based on a 40bit SSL. This strength is insufficient with nowadays technology.
- There is NO security advice for non-knowledgeable users
- This web-based system can be easily attacked by any Trojan program.

- The digital certificate, as shown in Figure 3-4, of Credit Mutuel is based on 512bit RSA.

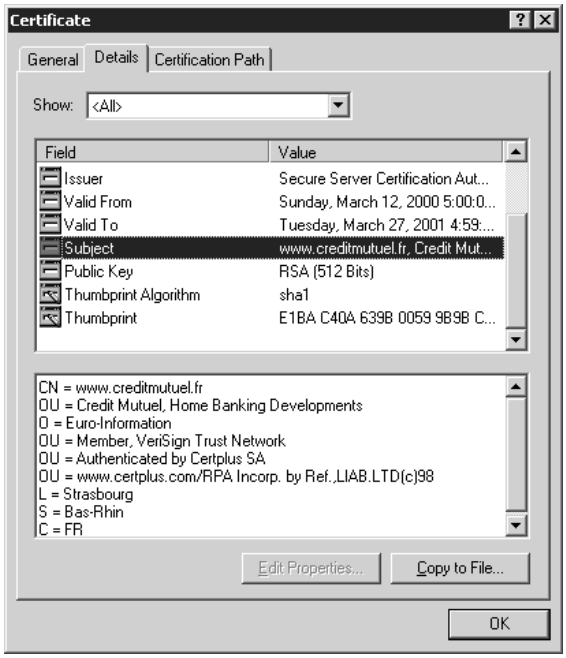

Figure 3-4: Digital Certificate of Credit Mutuel

4 Other Banks

During our research about Internet banking we also checked other banks online interface and found some obvious weaknesses or vulnerabilities [UBSweb, DeBweb, DrBweb].

4.1 UBS, Switzerland

4.1.1 Insecure download

UBS provides an application that can be used instead of their HTML interface and a so-called PIN-tool that handles the authentication process. The download page of UBS where the customers can download this software is not secured. There is no encryption and therefore also no certification for this page. It is an easy task for an attacker that has access to a Domain-Name-Server or a Proxy-Server to display a fake page to the user with modified software on it, which will help the attacker to get the necessary information to authenticate as a legitimate user.

4.1.2 Authentication Tool

In the UBS interface a user has to authenticate himself with a user-number, a password and a one-time PIN from a cancellation list. UBS has a tool, which handles the authentication process. It manages the cancellation list, which means it has a copy of it encrypted on the local hard-drive. The user has only to provide the password (over the keyboard) and the tool handles the rest of the authentication. This tool destroys all security that was added with the one-time PIN. The most likely place for password spoofing attack is on the users computer. It is unlikely that somebody cracks the 128bit encryption to get this information somewhere between the user and the bank. An attacker has now just to add a function to his spoofing tool that transmits beside of the password also the encrypted file from the hard-drive to him. Because he will know the password he can use the same tool to decrypt the cancellation list and authenticate himself as the legitimate user.

4.2 Deutsche Bank, Germany

4.2.1 Insecure download

The download page of Deutsche Bank where the customers can download several software tools and plug-ins for Internet banking is not secured. There is no encryption and therefore also no certification for this page. For possible attacks see 4.1.1.

4.2.2 Application Process

Deutsche Bank offers an application for a bank account on-line. The whole application process, while managing confidential information is not secure.

4.3 Dresdner Bank, Germany

4.3.1 Insecure download

In the authentication process of the Internet interface of Dresdner Bank a user need a disk with personal information (public key). The user needs special software to read this disk. The download page where the customer has to download the necessary software is not secured. There is no encryption and therefore also no certification for this page. Possible attacks see 4.1.1.

5 Security of SSL

All Internet Banking interfaces we found are based on SSL connections from the user's computer to the server of the bank. If anyone is able to break SSL they have broken all this interfaces. SLL itself is pretty strong, but implementations of the protocol may have flaws that make them vulnerable.

5.1 Router Discovery Protocol

An SSL connection can be attacked if on the computer the ICMP Router Discovery Protocol (IRDP) is enabled and the attacker is connected to the same local network. The fact that this protocol weakens an SSL connection was known when the protocol was defined. The possible attacks were even mentioned in the description of the standard (RFC 1256, Section 7). The problem is that Windows95 and Windows98 have IRDP activated by default [L0pht].

With activated IRDP protocol is a man-in-the-middle attack possible. An attacker can remotely add default route entries on a remote system by spoofing IRDP Router Advertisements. The default route entry added by the attacker will be preferred over the route obtained from the DHCP server. The attacker can then act as a proxy between the user and the end host. The user will think that he is connected directly to the host but in fact he is connected to the attacker. The attacker is connected to the host and is passing the information from the user to the host. In a SSL connection is the attacker even able to intercept the traffic unencrypted.

If the presence of an attacker on the local network can't be excluded, the IRDP protocol should be disabled on machines that are used for secure connections like Internet Banking. How IRDP can be deactivated on Windows95 and Windows98 machines can be found in the Microsoft Knowledge Base [MK41].

5.2 Invalid SSL Certificate Warning Bypass in NS-Communicator

In Netscape Communicator 4.05 up 4.72 exists a vulnerability in the manner how the program validates SSL certificates [CA-2000-05] [sf1188].

Besides the cryptographic protection SSL should make sure that the user is connected to the correct server. Because of the existing vulnerability in the NS-Communicator this can't be guaranteed. The problem is as follows: NS Communicator checks the certificate conditions correctly at the beginning of a SSL session it establishes with a certain web server. But, while this SSL session is alive, all HTTPS connections to the IP-Address of the server that was present when the session was established are assumed to be a part of this session. That means certificate conditions are not checked again. Instead of comparing hostnames to those of currently open sessions, NS-Communicator compares IP addresses. Since more than one hostname can possibly have the same IP address, there is a great potential for security breach. This behavior is not in compliance with SSL specification.

Here is an example of an attack using this vulnerability (taken from the 'SecurityFocus.com' homepage [sf1188]):

An attacker poisons a nameserver to redirect all connections to www.goodguy.com, normally 100.100.100.100, to 99.99.99.99, www.badguy.com. The attacker causes all normal http requests to return what they normally would on www.goodguy.com, even though a user attempting to contact www.goodguy.com hits www.badguy.com.

Upon getting a hit to www.badguy.com, the attacker causes an SSL connection to be established. This can be done by embedding a small image. The user may or may not get a warning about establishing a secure connection – this warning is on by default, although many users will choose to disable this warning. The attacker needs to use a legitimate SSL key, certified by someone listed as trustworthy (thwate.com, for instance)

The user can continue to shop to their hearts content, on the real site, as it's being proxied.

When the user decides to check out, it will attempt to establish an SSL connection to www.goodguy.com. Upon checking the ip address for www.goodguy.com, for establishing an SSL connection, it will note that an SSL connection already exists to its IP. The key, however, was issued to www.badguy.com. The SSL connection will be established, and by all indications appear to go to www.goodguy.com, when in fact it is to www.badguy.com.

Users of Internet Banking Services that use Netscape Communicator should update their browser to version 4.73 or install the patch for the older versions [NsSN].

6 Conclusion

In this report, we have studied different Internet banking solution schemes (HTML interface, Java interface and client program interface). We also describe two banking solutions provided by two different banks from different country: Credit Suisse from Switzerland and Credit Mutuel from France.

The security provided by these systems varies widely. In France regulations prohibit the use of strong encryption, therefore is the encryption of these solutions very weak (512bit RSA and 40bit SSL) while in Switzerland the level of security is strengthened (128bit SSL and 2048bit RSA). The French level of security is obviously unacceptable in the United States. Any Trojan program can easily take advantage of the weaknesses outlined in this report and hence retrieve confidential data.

The most likely attacks to any of the interfaces described in this document would try to attack the user's computer, because it is obviously the weakest link in the chain. Therefore is it very important that banks inform the users about the risks of Trojans and Viruses. It seems that most banks are not aware of the risk that such a program could reach the user's computer from a faked download page of the bank. We found only one bank with a secured download page.

To continue this work, we could focus on a client program based interface. The flexibility of this kind of program can be exploited to enhance the level of security on the client platform.

References

[CA-2000-05] CERT Advisory: CA-2000-05
Netscape Navigator Improperly Validates SSL Sessions
http://www.cert.org/advisories/CA-2000-05.html

[CSweb] Credit Suisse: Direct Net Webpage (Version from 05/22/2000)
http://www.en.credit-suisse.ch/directnet/

[DeBweb] Deutsche Bank: Login Banking Webpage (Version from 05/22/2000)
http://www.deutsche-bank-24.de/

[DrBweb] Dresdner Bank: Internet Banking Webpage (Version from 05/22/2000)
http://www.dresdner-bank.de/internetbanking/start_ib.html

[L0pht] L0pht – Heavy Industries (hacker organization): Advisory Database
Attackers can remotely add default route entries on the victims host
http://www.l0pht.com/advisories/rdp.txt

[MK41] Microsoft Knowledge Base: Disable IRDP
http://support.microsoft.com/support/kb/articles/q216/1/41.asp

[NsSN] Netscape Security Notes
http://home.netscape.com/security/notes/index.html

[CM] CyberMUT, Credit Mutuel
http://www.creditmutuel.fr

[RP99] T. Redhead and D. Povey. The Problems with Secure On-line Banking.
http://security.dstc.edu.au/papers/searcc98-bank/

[sf1188] SecurityFocus.com, Vulnerability Database, ID 1188
Netscape Navigator and Communicator Invalid SSL Certificate Warning By-pass Vulnerability
http://www.securityfocus.com/vdb/bottom.html?vid=1188

[UBSweb] UBS: Telebanking Webpage (Version from 05/22/2000)
http://www.ubs.ch/e/telebanking/classic.html