MASTER

Analysis of current and future phishing attacks on internet banking services

Hegt, Stan

Award date:
2008

Analysis of Current and Future Phishing Attacks on Internet Banking Services

by Stan Hegt

TECHNISCHE UNIVERSITEIT EINDHOVEN
Department of Mathematics and Computer Science

MASTER THESIS

Supervisors:
dr. B.M.M. de Weger (TU/e)
prof. J.J. Lukkien (TU/e)
M.R.A.M. Smeets MSc (KPMG)

May 2008

Management summary

This thesis discusses the results of our research into phishing in a Dutch internet banking context. We try to apply a structured approach in order to analyze the current situation and future prospects of phishing. We look at this phenomenon both from the attacker's and defender's point of view while maintaining a focus on technological issues.

First of all, we derive a definition of phishing. Our definition is broad in the sense that it includes attacks that may not be identified as phishing attacks by other researchers and phishing experts. Our definition abstracts from technology related issues, which allows us to reuse the definition for future attacks.

Web-based internet banking services exist for about a decade. In the last few years these services have become established technology in the Netherlands. Nevertheless, confidence in the security of internet banking services remains a critical issue. Phishing is the most apparent class of attacks on these services and the media are eager for news on these attacks. Consequently, banks have defensive strategies to counter phishing in order to avoid direct financial losses and probably even greater reputation losses.

In order to be able to assess the current defensive strategies and to see in which ways they can be improved we derived requirements on internet banking. These requirements address both security and non-security issues.

We identified the vulnerable entities in an internet banking architecture using a structured approach. This allows us to create a threat model which lists the most relevant phishing-related risks to internet banking. Key issues in our threat model are that we expect tampering with the customer's personal computer to be a major risk and we assume a powerful and well-organized adversary.

This assumption of a well-organized adversary turns out to be valid when we analyze the current modus operandi of phishers. Empirical evidence allows us to construct a model which enables us to identify and analyze phishing

attacks. We can distinguish three parts of a phishing attack, namely the lure, the hook and the catch. This distinction allows attackers to apply the principle of separation of concerns which enables illicit market places. Currently, we identify three major types of phishing attacks: dragnet phishing, real-time man-in-the-middle phishing and malware-based phishing.

In order to fight phishing attacks Dutch banks have employed defensive strategies. A major pillar in these strategies are the authentication schemes that are employed. Of these schemes we see two variants in the Netherlands, namely two-factor and two-channel authentication schemes. Both schemes lack a proper mechanism to protect the integrity of the data involved in internet banking activity. Hence, in order to ensure the integrity of the data transmitted to the bank a secure path from the customer to the bank is required. This leads to a focus on software security mechanisms such as firewalls and virus scanners that protect the customer's end systems and to a focus on SSL / TLS for the protection of the connection to the bank. Unfortunately, these defensive controls have serious deficiencies.

As a consequence of these deficiencies we can still identify main points where phishing attacks can evolve. We conclude that this evolution will take place especially in the lure and hook of a phishing attack. We think that the developments of future attacks will be bipartite. On the one hand we will see the employment of new technology (broadening) and on the other hand we will see more sophisticated exploitation of current technologies (deepening). Examples of broadening future attacks are vishing and man-in-the-mailclient attacks. Examples of deepening future attacks are spear phishing and man-in-the-browser attacks.

As a solution for the deficiencies in current defensive strategies and as a protection against future phishing attacks we propose an enhanced defensive strategy. We propose a signing application running in a trusted computing base only available under direct physical control of the customer that requires a customer to digitally sign every crucial internet banking activity and the relevant information involved in this activity. Our solution does not require a secure path from the customer to the bank and hence relies less on software security and SSL / TLS controls.

Preface

This master thesis is a product of the research I have conducted for my graduation project from September 2007 until May 2008. This graduation project is the final part of my master education on Information Security Technology, a specialization program of the Computer Science & Engineering education at Eindhoven University of Technology. The research was carried out at KPMG ICT Security & Control in Amstelveen, the Netherlands. This is a business unit of KPMG IT Advisory which specializes in advisory and audit services on information security issues.

I truly enjoyed the working environment at KPMG ICT Security & Control. This business unit is full of inspiring and knowledgeable colleagues who excel in verbalizing security issues. Moreover, these colleagues were willing to share their contacts in the banking industry with me, which provided me with lots of valuable resources. Unfortunately, because of the extremely inconvenient traveling distance from my howetown Den Bosch to Amstelveen I decided not to stay at KPMG ICT Security & Control after the completion of my graduation project.

Furthermore, I praise the Information Security Technology education. This master program is an exquisite choice for students who are captivated by information security issues. For me as a whitehat hacker this education was a perfect and interesting way to support my practical skills with broad theoretical and academical knowledge. Despite of the small numbers of students that enroll in this education I really hope the motivated instructors continue with this contribution to the Dutch information security industry and research. The founding of the Kerckhoffs Institute is definitely a major achievement.

Last but not least I would like to express my word of thanks to a number of people from KPMG who facilitated my research. First of all I would like to thank Arjen van Zanten, partner at KPMG ICT Security & Control. Arjen approved my position at his business unit and he provided me with all the freedom required to realize a research project that met my insights. Arjen, I wish you strength in dealing with your illness. Furthermore, my

gratitude goes out to Peter Kornelisse, director at KPMG ICT Security & Control. Peter arranged meetings with contacts in the banking industry, which considerably contributed to my research. I admire Peter for his social skills and I would like to thank him for the genuine interest in my research despite of his busy schedule. Moreover, I appreciate all the support of Erwin Hansen, senior manager at KPMG ICT Security & Control. He assisted me during the application process and supervised my research at an organizational level. I also owe much gratitute to Marc Smeets, junior advisor at KPMG ICT Security & Control. He supervised my research in a convenient manner. I would like to thank him for the trust and freedom that I really appreciated while conducting my research. Marc, you are a warm personality and I wish you all the best in your carreer.

Moreover, my gratitude goes out to some people at the Eindhoven University of Technology. My thanks goes out to Bart Jacobs for willing to participate in the assessment committee of my graduation project and for arranging a very interesting meeting with the Radboud University and Rabobank. Furthermore, I would like to thank Johan Lukkien and Benne de Weger for supervising my graduation project. Together Benne and Johan form a great combination. They provided me with keen and sharp criticism on my thesis. Benne and Johan, I really appreciate your effort and the way you supported my graduation project.

Finally, I would like to thank the experts from the banking industry that I interviewed. They provided me with lots of interesting information on internet banking and phishing.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 A quick history of phishing

Phishing is far from a new phenomenon. About a decade ago the first phishing incidents involving AOL account theft were reported. A few years later fake Hotmail login sites were a popular method for identity theft. Fortunately, the effects of these attacks on the victim were above all of an annoying nature and mostly did not yield substantial financial consequences.

Meanwhile, as the internet evolved, the services offered became more mature. A few years ago financial institutions started offering internet banking and online payment systems. By now, these systems have been widely accepted and it is estimated that about 70% of the Dutch internet users make use of internet banking systems [76]. Consequently, attacks on these systems became interesting for organized crime. In contrast to early phishing attacks, which were mostly only annoying to the victim, the phishing phenomenon had turned into a serious felonious business mainly targeting financial services.

By 2004, phishing was recognized as fully industrialized in the sense of economy of crime: On the black market off-the-shelf components for ready-made phishing attacks were for sale [23]. Meanwhile, financial services have become the prime victim of phishing attacks. By now, it is estimated that the annual direct financial loss in the United States of America because of phishing attacks is 2.8 billion dollar [13][63]. Some researchers consider this amount to be exaggerated but a bare minimum annual loss of 350 million dollar seems very plausible [68]. Reliable indications of indirect losses are not available but these may be much higher.

In the last year we have seen the first indicators that a new generation of technically sophisticated phishing attacks is about to arise. In June 2006

the first phishing attack that involved cross site scripting was carried out [69]. The attackers sent a specially crafted URL to several PayPal users. Exploiting PayPal's lack of input validation the users that visited this URL encountered a page on the legitimate paypal.com domain saying that their account had been suspended due to abuse. When a victim clicked the button on this page, he was redirected to a login page on a host controlled by the attacker. Note that the difference with traditional phishing attacks is that the failure message on the first page was on the domain of PayPal. Hence, it was part of a legitimate SSL connection with PayPal.

In March 2007 customers of the ABN Amro bank fell victim to a real-time man-in-the-middle attack. Using a mass-mailing that propagated malware the attackers were able to infect thousands of computers. Once infected, customers that tried to visit the ABN Amro internet banking site were redirected to a malicious website. This website effectively performed a man-in-the-middle attack in real time which allowed the attackers to successfully circumvent the two-factor authentication deployed by ABN Amro.

Clearly, from this timeline we can conclude the phishing phenomenon is evolving from both a technological and a organizational viewpoint. In this study we will explore current phishing attacks and defenses from the technological viewpoint and we will thoroughly examine attack vectors and matching defenses that we can expect to enter the phishing scenery in the coming years.

1.2 Towards a definition of phishing

Among phishing researchers there is little consensus about what phishing is and there is an even greater debate on what phishing is not. Almost every textbook, academic paper or research report uses its own definition on phishing. We will derive our own definition of phishing in order to settle our view on what is phishing and what is not.

Let us first have a look at some sample definitions from prominent works on phishing:

1. *Phishing: A form of social engineering in which an attacker, also known as a phisher, attempts to fraudulently retrieve legitimate users' confidential or sensitive credentials by mimicking electronic communications from a trustworthy or public organization in an automatic fashion. [56]*

2. *Phishing is a form of internet scam in which the attackers try to trick consumers into divulging sensitive personal information. The techniques usually involve fraudulent E-mail and web sites that impersonate both legitimate E-mail and web sites. [85]*

3. *Phishing is an attack in which victims are lured by official looking email to a fraudulent web-site that appears to be that of a legitimate service provider. [33]*

4. *In phishing, an automated form of social engineering, criminals use the internet to fraudulently extract sensitive information from businesses and individuals, often by impersonating legitimate web sites. [71]*

5. *Phishing attacks. These are attacks in which, typically, the victim is deceived to give out secret information such as passwords or other information enabling access to a given resource. [55]*

6. *Phishing, the practice of directing users to fraudulent web sites. [39]*

Firstly, a striking point is that a number of the phishing definitions described here are ostensive definitions. Namely they point at examples of aspects of phishing and expect the meaning of the term phishing to become established in this way. However, such case-based reasoning only introduces a vague and subjective understanding of the concepts of phishing.

Moreover, the aforementioned definitions introduce a dilemma. On the one hand we see definitions that have a very strict view on phishing (for example definitions 3 and 6). On the other hand there are researchers that consider phishing in a broader context (for example definitions 1 and 5). As

was already pointed out in the introduction, phishing scams are evolving. Consequently, these scams can no longer be considered in a narrow sense. A definition of phishing that remains adequate and manageable reckons with future developments.

There is simply too much controversy to make it possible to introduce a universally accepted descriptive definition [1] which suits all existing literature on phishing. All things considered, this leaves us with the need to introduce a stipulative definition[2]. As discussed this definition should be a broadened view on phishing that allows for future evolution of the term.

Consequently, an inevitable result of our stipulative definition is that it will conflict with some of the narrower definitions of phishing that can be found in literature. My argument for this is that a definition of phishing should not focus on the technology being used, but rather on the methodology of the scam. I advocate this by pointing at *skimming*, an offline scam that has many analogies to phishing. If one defined skimming as copying magnetic stripe data from payment cards, the utility of this definition would be very limited. Namely, this focus on technology renders the term inapplicable upon introduction of payment smartcards or RFID payment cards. A much better definition of skimming would be *theft of payment information used in an otherwise legitimate transaction*. In this manner the definition reckons with evolution of technology and the accompanying crime. Another example is *safecracking*. If several decades ago one described this term using specific locksmith technology the term would have rendered inapplicable once digital locks came into use on safes. Likewise, a phishing definition should not focus on current technology such as e-mail, WWW and webbrowsers. It is likely that these technologies will once be outdated and that their successors will be incorporated into phishing attacks.

Then what are the essential elements of a phishing attack? First of all there is the spoofing element. In the aforementioned definitions this is expressed by words such as mimick, deceit, trick and official looking. We define spoofing in a phishing context as a deceit that tricks a party into believing that he is involved in a genuine transaction with another legitimate party, while actually a malicious entity influences the transaction.

Furthermore, all definitions address an aspect that can be summarized by the term leakage of confidential information. However, only a few definitions mention that this information is actually stolen with criminal intent. This is an important difference. Phishers do not catch data that is accidently

[1] A definition of the meaning that a term bears in general
[2] A definition introduces a meaning to a term that is already in use

leaked, but they steal the data with premeditation. Note that the term stealing can be ambiguous in a digital context. In our definition, stealing covers both the copying and interception of data.

Note that phishing is not only the act where one deprives confidential information. It is a *process* that also incorporates the propagation of a phishing lure, setting up a hook and exploiting the information once it is stolen. None of the aforementioned definitions explicitly address this.

Only one of the given definitions mentions explicitly that phishing targets electronic communications. In my opinion, this is an essential ingredient of a phishing attack. Attacks on offline real-world transactions are not considered phishing.

Altogether, we define phishing as:

Phishing is the process in which an adversary attempts to steal and exploit confidential information by leading a human being into believing to be involved in an electronic transaction with a legitimate party while actually the adversary exerts influence on this transaction.

In the famous textbook Phishing Exposed it is stated that phishing attacks are essentially man-in-the-middle attacks[3][59]. Note that this is subtly different from our definition: our definition allows for phishing to be carried out by means of man-in-the-middle attacks but does not rule out attacks where an attacker spoofs one institution and subsequently impersonates the victim at another institution using the stolen information. In the latter case the attacker did not exert influence on communication between the victim and the institution that was mimicked and hence such an attack should not be considered a man-in-the-middle attack.

Finally, we would like to clarify the relation between the terms phishing and identity theft. These terms have a common ground: Phishing is a mechanism that can be incorporated to perform identity theft. However, it is important to note that the majority of identity theft crimes are committed in an offline context (i.e. not related to the internet) [47] and are thus not related to phishing.

[3]An attack in which an attacker is able to read, inject or modify messages in a communication between two parties without either party knowing that the link between them has been compromised.

1.3 Demarcation of the study

1.3.1 Problem description

Phishing is a generic term for a proliferation of information theft scams. Many variations of attacks are carried out in the wild and an even greater number of countermeasures is invented and deployed. However, it is doubtful whether these defensive mechanisms offer sufficient protection against current attack methodologies. Not to mention whether these defenses will be of any help against future attacks or that they target the symptoms instead of the root causes of phishing attacks. A lack of understanding of the structure of the phishing phenomenon is the foundation of the problem that the technological clash of future phishing attacks and their countermeasures is uncertain.

1.3.2 Research questions

The problem description leads to the principal research question of this study:

How can a structural approach support analyzing the technological future prospects of phishing attacks and defenses?

The principal research question will be answered by means of a subdivision into four more detailed questions.

- *Q1: In what ways is it possible to structure the methodology of phishing techniques and defenses?*

  Phishing is a dymanic phenomenon which, in the course of time, has evolved into a broad spectrum of offensive techniques and corresponding defensive mechanisms. For example, malware has come into use in several phishing attacks [19]. Moreover, the set of defenses ranges from back-end transaction anomaly detection systems to browser toolbars. As a consequence of this evolution researchers have conflicting views on what the boundaries of phishing are. Obviously, such an inconsistency complicates information exhange. Hence, an adequate definition of the term phishing is desirable. For this, we use the definition given in the previous section. Furthermore, a structural approach is required to construct a framework which enables uniform identification and analysis of present and future phishing attacks and defenses.

- *Q2: How will phishing attacks presumably evolve in the future?*

  For risk management purposes a reliable forecast of the evolution of phishing attacks would be extremely valuable. However, such a prophecy quickly leads to vague estimates. In order to prevent this, one can use the results from the previous research question. By applying the structured phishing framework to the current modus operandi of phishers it is possible to identify which branches leave headroom for technological sophistication. Combined with an extrapolation of trends this yields an interesting and realistic view on the future of phishing.

- *Q3: To what extent will currently deployed anti-phishing mechanisms protect against future attacks?*

  Dutch internet banking services have spent considerable amounts of money on phishing defenses such as two-factor and two-channel authentication schemes. Not only do these front-end systems require high investments, their large-scale deployment programs also make them rather static. Hence, these mechanisms should provide protection against phishing attacks for the coming years. By analyzing current defensive mechanisms and projecting these on the foreseeable future a conclusion can be drawn on the validity of these expectations.

- *Q4: Which defensive techniques are required to protect against the future prospects of phishing attacks?*

  Obviously, evolving attacks pose new requirements on countermeasures. Instead of structurally lagging behind, formulation of these requirements enables anticipation on future techniques. Furthermore, reviews of state-of-the-art technologies give insight into the question of whether there are techniques available that meet or come close to these requirements.

Scope

This research mainly considers phishing in an internet banking context. This focus has some subtle implications which turns phishing in an internet banking context into a different scenario than with other targets of phishing, such as eBay and PayPal. For example, banks have to protect assets of a higher value than eBay and PayPal do. Moreover, banks have a much closer relation to their customers. Their business models already incorporates contact to customers via postal service or other non-internet channels.

Furthermore, the focus of this research is on technological aspects. Therefore the research results of this study are primarily intended for technically oriented people that have affinity with digital crime. Nevertheless, one can not study the complex and diverse topic of phishing having blinders on. Consequently, this research will briefly mention non-technological aspects with respect to the broad context of phishing.

Lastly, when referring to the term *future* in Q2, Q3 and Q4 a time span of 3 years is considered.

1.3.3 Research methodology

Our research will mainly be an exploratory investigation of the phishing phenomenon. Partly, a literature review will be conducted. Phishing is an actual and popular topic among academic and commercial researchers. Consequently, this yields an overwhelming lot of information on a range of aspects of phishing. It is important to also examine the phenomenon from an insider view. In order to do so a discussion will be raised with representatives from parties who are directly involved in combating phishing, in particular with phishing experts from banking corporations. Furthermore, the phishing scenery will be explored from the perspective of the attackers by the development of a technically sophisticated phishing attack which exploits a futuristic attack vector.

As a first stage of the research a framework to identify and analyze phishing attacks is constructed. This framework will be based on information from literature and on current phishing attacks as empirical evidence. Subsequently, the framework will be reviewed by phishing experts from various parties that are involved in fighting phishing. Furthermore, based on literature, empirical evidence and expert opinions an overview of the phishing phenomenon will be given. The framework and this context overview together form a guideline for an analysis of current countermeasures. Up to here, the research will yield enough information to perform a risk analysis of future phishing attacks. As a practical support for this theoretical risk analysis a practical exploration of state-of-the-art phishing attacks and defenses will be done.

Man-in-the-Browser case-study

Parallel to the methodology described above, answers to questions Q2, Q3 and Q4 from our research questions will be given with the help of a case study of Man-in-the-Browser attacks. A practical attack of this technique for major Dutch internet banking services will be developed. The development process of this attack is documented to assist the answering of Q2. Q3

is answered by analyzing the impact of popular virus scanners, anti-malware products and personal firewalls on this attack. Finally, new security mechanisms and enhancements to current mechanisms are proposed in order to answer Q4.

1.3.4 Added value of this study

This study realizes a number of contributions to both the academic knowledge and to the industry, in particular internet banking services. The most important contributions are summarized by this list:

- We present a definition of phishing which is compared against existing definitions of phishing. Of course, we do not regard our definition to establish a uniform view on the term phishing. However, it might contribute to the understanding that this term is often ambiguously used in both the academic and business world. *(Chapter 1)*

- We identify requirements for internet banking systems to which phishing defenses should comply. These requirements provide a scheme to assess the completeness and adequacy of both individual and aggregated anti-phishing controls. *(Chapter 2)*

- We derive a new threat model that gives insight into the weaknesses of internet banking from a phishing perspective. This model supports in understanding the basic threats that enable phishing. Moreover, this threat model gives the opportunity to identify threats that phishers may exploit in the near future. Lastly, it aids in the assessment of the completeness and adequacy of anti-phishing controls. *(Chapter 2)*

- We devise a framework that structurally describes the modus operandi in phishing attacks. This framework allows for the identification and demarcation of phishing attacks. Furthermore, it gives insight into possible chokepoints in phishing attacks that anti-phishing controls might counter. *(Chapter 3)*

- We present an evaluation of the current defensive strategy against phishing in the Netherlands. Our evaluation points out shortcomings which could support decision making at financial services. *(Chapter 4)*

- We demonstrate the directions in which we think phishing will evolve. For example, we demonstrate the practical feasibility of man-in-the-browser attacks and we discuss a class of man-in-the-mailclient attacks that has never been documented before. *(Chapter 5)*

- We propose an enhanced defensive strategy which provides a remedy for the deficiencies in current defensive strategies. Moreover, this defensive strategy provides protection against future phishing attacks. Not only do we address the pratical issues but we also discuss the theoretical concepts of our defensive strategy. *(Chapter 6)*

1.3.5 Information resources

Unfortunately, information on phishing coming from parties who are directly involved in the phenomenon is narrow. In general consumers and corporations are unwilling to disclose information on phishing and related incidents because of apprehension for legal, branding and financial consequences. Moreover, for obvious reasons of self-protection criminals do not expose their illicit activities. Fortunately, there are a number of autonomous parties who perform research into the phishing phenomenon. The following are an overview of the classes of information resources used for this research.

Literature resources

Several general-purpose commercial research companies examine the phishing phenomenon. One of the most prominent in this field is Gartner[4], which publishes commercial research results on phishing on a regular basis. Similar research is performed by the Tower Group[5] and The Ponemon Institute[6].

Phishing is also a popular topic in the academic research world. For example, Richard Clayton et al. from the Computer Laboratory of the University of Cambridge (UK) have done some valuable research into the world of phishing. Their findings are published in several papers and on their blog[7]. Another notable academic contributer of phishing research material is Markus Jakobsson, an associate professor at the Indiana University School of Informatics. Jakobsson published several papers on phishing [45][57][58] and was a co-editor of the book 'Phishing and Countermeasures' [56].

Non-literature resources

Furthermore, phishing-specific research institutes exist. The most notable one is the Anti-Phishing Working Group[8], whose 2600 members report and discuss phishing trends. The Anti-Phishing Working Group is also responsible for organizing the APWG eCrime Researchers Summit, an academic conference focusing on phishing and other forms of online crime. Other

[4] `http://www.gartner.com`
[5] `http://www.towergroup.com`
[6] `http://www.ponemon.org`
[7] `http://www.lightbluetouchpaper.org`
[8] `http://www.apwg.org`

interesting phishing-specific resources are PhishTank[9], MillerSmiles[10] and Castlecops[11]. These organizations are run by volunteers and excel in collecting and evaluating reports on phishing incidents. Their work is of great value for the creation of phishing blacklists.

For this study we have interviewed a number of phishing experts from Fortis Bank Nederland, ING Global CERT and De Nederlandsche Bank. These experts provided far-reaching and actual insights into the rapidly evolving scenery of phishing. Thanks to them, we got access to information that is too recent or too confidential to appear in literature or whitepapers. In this thesis we frequently refer to these interviews for supporting a statement. Please note that these references are made in an anonimized manner. That means, no reference is given to the specific expert or bank for each expression made by these experts.

[9] `http://www.phishtank.com`
[10] `http://www.millersmiles.co.uk`
[11] `http://www.castlecops.com`

1.4 Outline

This thesis is structured as follows:

After this introduction we discuss internet banking in Chapter 2. Some facts and figures on internet banking are provided and the need for confidence in internet banking security is stressed. Moreover, in this chapter requirements on internet banking are derived. These requirements form a framework to assess the current defensive strategies and to see in which ways they can be improved These requirements address both security and non-security issues. Additionally, this chapter provides a threat model that lists phishing-related threats in an internet banking context.

Subsequently, in Chapter 3 we analyze the modus operandi of phishers. In this chapter a model to identify and analyze phishing attacks is derived. We show how current phishing attacks consist of distinct parts and how these parts are interconnected. Finally, we analyze the most prevalent phishing attacks.

In Chapter 4 we present an evaluation of the current defensive strategy against phishing in the Netherlands. First, front-end solutions are addressed. Second, we deal with back-end controls. This chapter is closed by an evaluation that summarizes the strengths and weaknesses of the current defensive strategy employed by Dutch banks.

Chapter 5 demonstrates future development directions of phishing attacks. It is shown in which parts of phishing attacks we expect the most development in the near future. We discuss the practical aspects of spear phishing and vishing. Moreover, the feasbility of man-in-the-browser attacks is demonstrated. Finally, we document on a class of attacks called man-in-the-mailclient attacks that is yet unexplored.

Next, in Chapter 6 we propose an enhanced defensive strategy as a solution for the deficiencies in current defensive strategies and as a protection against future phishing attacks. First, we discuss the theoretical concept of such a defensive strategy. Subsequently, considerations for practical implementations of this concept are discussed.

Finally, in Chapter 7 we present our conclusions of this research and recommendations to internet banking services.

Chapter 2

The internet banking context of phishing

In this chapter the internet banking context of phishing is introduced. We start off with a short summary of the facts and figures of internet banking. It is stressed how important the confidence in security is for the reputation of internet banking. Subsequently, requirements for internet banking are derived. Any anti-phishing control should meet these security and non-security requirements. The last part of this chapter deals with the development of an internet banking threat model that helps us to analyze phishing.

2.1 Facts and figures

2.1.1 Genesis of internet banking

The first Dutch bank that offered home banking services using personal computers was Postbank, which introduced the Girotel application in 1986. In the very beginning an external computing device called Viditel was required, which had to be connected to both a telephone line and the user's monitor. Using this system a user could initiate transactions locally and transmit these to the bank over a telephone line in a batch.

Eventually, the Viditel device was replaced by a fat-client [65] software application which had a similar operating procedure. Other banks followed with similar applications such as Fortis Bank's PC Banking and ING Bank's Interactive Banking. Not only competitiveness amongst banks was a motive to introduce these home banking applications. For banks major benefits of these remote banking services are the reduced personnel and administrative costs.

In 1997, 2 years after the global introduction of internet banking by Security First Network Bank (USA), Kas-Netbank was the first Dutch bank to introduce domestic transactions via the World Wide Web. Rabobank was the first major bank that followed. Soon, almost all major Dutch banks offered internet banking services that could be accessed from any internet-connected computer that had a web browser installed.

Nowadays, internet banking has become established technology. All major Dutch banks offer advanced internet banking services that offer domestic and international payment transactions and alerting services using e-mail or SMS. Some banks take remote banking even further. For example, Rabobank has enrolled Rabo Mobielbankieren [1], an SMS text messaging based banking service, and is experimenting with internet banking using television and remote controls.

The establishment of internet banking is also expressed by its large user base. In 2006, 68% of all internet users regularly used internet banking services [5]. For the coming years a considerable growth in the number of users of internet banking is expected. After that the number of users will likely remain stable [7].

[1] http://www.rabomobiel.nl

2.1.2 Confidence in internet banking security

Right from the introduction of internet banking in the Netherlands security of these systems has been subject for debate. Fortunately, all major Dutch banks introduced strengthened authentication and security measures, in contrast to many other banking systems worldwide that used trivial username and password protection.

Especially so-called two-factor authentication systems became popular right from the introduction of internet banking. Using such a system, initiation of an internet banking session or a transaction requires access to an additional device (often called security calculator), which can generate temporal access codes. Postbank was one of the few banks that deployed a different system (two-channel) with their Girotel Online internet banking application, which incorporates TAN-codes (Transaction Authorization Number) which serve as one-time authentication codes. These codes are distributed via a non-internet channel, first by postal service. Later, in 2004 with the introduction of the free MijnPostbank internet banking application, TAN-codes could also be transmitted to customers via SMS text messaging. Up to now Postbank has maintained this two-channel system. All other major banks have refined their two-factor authentication schemes. Both two-factor and two-channel authentication systems will be extensively discussed in Section 4.1.2.

In 2000 it was demonstrated how media are eager for insecurities in remote banking systems. Hackers from the infamous computer security group Klaphek showed a scenario in which ABN Amro's home banking application HomeNet was under attack. By examining popular mailing lists the attackers extracted email addresses from ABN Amro customers that asked questions about HomeNet on these mailinglists. Subsequently, the attackers sent these customers a counterfeit update for the HomeNet application which in fact contained the Trojan horse Back Orifice[2]. Since the home banking application lacked any form of integrity protection on the file that contained the batch of transactions to be sent, the attackers could then modify all transactions that were included in this batch. Although the attack was rather trivial it received massive attention on Dutch national television (including the RTL 4 Nieuws) and ABN Amro felt forced to send a letter to all of its customers[3].

[2]Around 2000 Back Orifice was a popular remote administration tool. Because of its stealthy features it became especially popular among malicious hackers. See `http://www.cultdeadcow.com/tools/bo.php` for more information.

[3]The letter can be found at `http://tweakers.net/nieuws/13115/abn-amro-stuurt-brief-over-homenet.html`.

Recently, media attention for alleged internet banking insecurities seems to have proliferated. Popular journals, magazines [76], websites[4] and even the monthly of the Dutch consumer union [6] all addressed internet banking security and in particular phishing.

Whether influenced by all this media attention or not, according to research from De Nederlandsche Bank one third of all internet users that do not use internet banking consider security issues a barrier for using internet banking services [7]. Strikingly, according to the same research elderly have not a significantly different perception of internet banking security than younger people who are more acquainted with modern technology. Furthermore, customers that do use internet banking services judge the security of these services only barely satisfactory.

Accordingly, it is not surprising that interviews with phishing experts from majors banks showed that the public reputation of internet banking is a major issue for these banks. This focus on confidence in internet banking is not misplaced: Our interviews with experts also demonstrated that in case of a major security incident (e.g. phishing), for capacity reasons a fall-back arrangement to return to traditional offline banking is not feasible. Evidently, internet banking has become a vital factor in our modern economy.

[4] For example see `http://www.webwereld.nl/ref/newsletter/48973`.

2.2 Requirements on internet banking

A list of requirements apply to internet banking applications. In the scope of our research into phishing we are especially interested in requirements that deal with security. However, any measures taken to achieve these security requirements must of course comply with other aspects such as costs and user-friendliness. Accordingly, we also treat such requirements that are not directly related to security. Finally, legal aspects on internet banking security and phishing are considered.

2.2.1 Security requirements

In order to aim for correctness and completeness of our security requirements we derive these from a document on riskmanagement principles for electronic banking created by the Basel Committee on Banking Supervision [2]. The purpose of the committee is to encourage convergence towards common approaches and standards and its members are central banks and federal monetary institutions from 11 major countries from the European Union and the United States. For example, De Nederlandsche Bank is positioned in it. In the aforementioned document this committee addresses seven issues that security controls in an electronic banking environment should deal with:

- Authentication
- Non-repudiation
- Data and transaction integrity
- Segregation of duties
- Authorisation controls
- Maintenance of audit trails
- Confidentiality of key bank information

This list of key issues would not suffice as a list of security requirements. First of all, the issues of *Segregation of duties*, *Authorisation controls* and *Maintenance of audit trails* are measures and no goals or requirements by itself. Furthermore, the committee discusses the *Authentication* issue unilateral from a bank's point of view: Only authentication of the customer is considered. The document does not address authentication of the bank's systems to the customer.

Hence, transformation of these issues into requirements is required. The

issues of *Non-repudiation*, *Data and transaction integrity* and *Confidentiality of key bank information* can be compiled straightforward. We expand the *Authentication* issue with authentication of the bank's systems to the customer. Furthermore, we interpret the means of *Maintenance of audit trails* as a requirement of accountability. Finally, for completeness reasons we add *No-intrusion* and *Freshness* requirements.

Before we present the list of security requirements let us clarify some terminology used in the following requirements. A *session* is considered as a dense series of communications that has a clear start in the form of a login action of the user and a corresponding termination action either by a logout action of the user or a disconnect from the banking systems. Within a session *information* can be exchanged, such as credentials, transaction instructions and balance information.

Let us consider the requirements which are directly related to security of an internet banking application:

- *No-intrusion.* An attacker cannot inject information in a legitimate internet banking session. As a consequence, if an attacker wants to impersonate a customer he has to do so from the very beginning of a session.
- *Authenticity.* Authenticity in an internet banking context is a mutual issue. Hence, it can be refined into two separate requirements:
  - *Server authentication.* The banking application must proof its identity to the customer at the start of a session.
  - *Customer authentication.* The customer must proof its identity to the banking systems at the start of a session.
- *Freshness.* All information exchanged during an internet banking session must be valid exclusively during that unique session. Consequently, captured credential or transaction information cannot be replayed outside a session or in a different session.
- *Confidentiality.* All information that is exchanged in an internet banking session must be exclusively available to the customer and the banking systems.
- *Integrity.* All information exchanged between bank and customer must arrive at the receiver's side as intended by the sender. Consequently, an attacker cannot modify information that is exchanged in an internet banking session.[5]

[5]Note the difference between integrity and no-intrusion. In our definition integrity

- *Non-repudiation.* The customer and the bank must be able to verify that the sender and receiver of information exchanged in an internet banking session are indeed the parties who claimed to have received or sent the information. This establishes a verifiable binding between this information and the identity of the sender or receiver. Hence, the sender of information is provided with proof of delivery and the receiver of information is provided with proof of the identity of the sender. One of the consequences of non-repudiation is that a customer cannot plausibly deny after sending transaction instructions.

- *Accountability.* The bank must be able to trace utilization of internet banking functionality (e.g. logging in and initiating a transaction) to a unique customer.

Note that the third term of the CIA triad of information assurance (confidentiality, integrity, availability) is explicitly omitted from the list of requirements. Achieving availability of internet banking applications over a best-effort system such as the internet is a Utopia. Fortunately, customers can fall-back to offline traditional banking in case internet banking is unavailable (up to a certain capacity, off course).

2.2.2 Non-security requirements

Requirements that are not directly related to security are relevant in a phishing context as well. Namely, these requirements should not be broken by phishing counter measures. Moreover, these requirements should not break our security requirements. Let us consider non-security requirements that are relevant to phishing:

- *Roaming.* The internet banking application must support mobile usage. Hence, customers must be able to use the internet banking application on every capable end-system. For example, customers must be able to use internet banking at home, work or at public internet computers in hotels with the same level of security.

- *User-friendliness.* Every person who can operate a computer and who is familiar with the internet must be able to use the internet banking application. This means that the system is also accessible for disabled and technically less oriented people.

- *Low-cost.* The costs of an internet banking system and the set of anti-phishing solutions that it is accompanied by must be as low as possible while still satisfying all other requirements. In particular, this implies

targets the modification of information while no-intrusion targets the injection of information.

that the costs of implemented anti-phishing solutions are in proportion to the phishing risks to which the internet banking system is exposed.

2.2.3 Legislative and compliance aspects

In addition to the requirements described in the preceding sections internet banking services (and thus also any incorporated anti-phishing measures) must comply with legal aspects and guidelines imposed by formal organisations. In the Netherlands De Nederlandsche Bank (DNB) is responsible for safeguarding domestic financial stability. Since financial stability depends on secure and reliable payment systems, internet banking services are under the supervision of De Nederlandsche Bank.

Strikingly, De Nederlandsche Bank does not impose any regulations or directives specifically on internet banking services[6]. Then what does De Nederlandsche Bank do as a supervisor on internet banking? Most importantly, DNB supervises the compliance to the guidelines and recommendations expressed in *Regeling Organisatie en Beheersing* [8]. This set of regulations contains a section dedicated to information technology, which prescribes rules and best practices for IT operations and control on these operations. However, note that these regulations do not contain any rules that specifically target internet banking and least of all technology related to phishing. This statement also holds for the recommendations in the Basel II Accord by the Basel Committee on Banking Supervision.

This is in contrast to the situation in the United States, where the Federal Financial Institutions Examination Council (FFIEC) has formulated guidelines for authentication for internet banking services [10]. The FFIEC expects banks to adopt some form of two-factor authentication to verify the online identity of their customers. However, it should be noted that the term *two-factor authentication* might be misleading in this context since the guideline also considers IP address restrictions and out of band authentication (two-channel) to be two-factor systems.

[6]This statement is explicitly expressed at `http://www.dnb.nl/dnb/home/over_dnb/veelgestelde_vragen/overige_vragen/nl/46-150354-64.html`.

2.3 Threat modeling

A threat model is a description of a set of security aspects of a system. In the context of our research the system to be examined is an internet banking application. In order to describe the security aspects of this system an exploration of potentially vulnerable entities is essential. Moreover, a threat model demarcates the scene: It is important to describe which types of threats are not considered. A valuable threat model is one which only includes realistic attacks and which thereby is an indicator for which types of attacks to prevent. After all, protection against an adversary who controls the universe is implausible and senseless.

A phishing threat model is essential in order to understand the notion of vulnerabilities that bring forth phishing attacks. Hence, it allows one to invent countermeasures that target the fundamental concepts of phishing instead of specific vulnerabilities. Moreover, a threat model gives insight into possible phishing attack vectors that are not yet being exploited but that may be in the future.

2.3.1 Existing threat models

Many threat models have been developed for various information security contexts. Let us explore whether these models are adequate to describe the threats related to the phishing phenomenon.

The Internet Threat Model

The Internet Threat Model was described by Eric Rescorla [77] as an attempt to model the threats that SSL / TLS [40] protects against. Rescorla utters some sweeping statements concerning this threat model. First of all, it is stated that *"designers of Internet security protocols typically share a more or less common threat model"*. Second, Rescorla makes two controversial assumptions. He assumes that *"the actual end systems that the protocol is being executed on are secure"*. This assumption is supported by his statement that *"users can expect that their own machines have not been compromised"*. On the other hand, he considers the communication channels to be compromised by an active adversary: *"We assume that the attacker has more or less complete control of the communications channel between any two machines. He can certainly inject packets into the network with arbitrary address information, both for the sender and the receiver, and can read any packet that is on the network and remove any packet he chooses"*. It is questionable whether these assumptions are realistic with respect to internet security and, more specific, phishing.

The assumption of the compromised communication channel seems reasonably plausible. Malicious employees of Internet Service Providers or various government authorities have complete control over the communication means. Furthermore, routers and switches suffer from numerous vulnerabilities that can be exploited by malicious hackers.

The other assumption, on the security of end systems can be considered incorrect in the context of the modern internet era. Home user's computers are the target of vast amounts of viruses, drive-by exploits and other online dangers. The emergence of bot-nets that control large numbers of computers shows that end systems cannot be considered secure[30]. Even anti-virus manufacturers lag behind (see Section 4.1).

Thompson threat model

This threat model is named after the work of Ken Thompson. Already in 1984 Thompson showed that one cannot trust any software and bootstrap code that run on the user's hardware [87]. He demonstrated this by implementing a compiler-compiler that inserts a Trojan horse. The basis of this model is Thompson's suggestion that source code verification or any other form of inspection will not protect one from malicious code.

Thompson's model has very impractical and rigorous consequences when applied to phishing: Implementing the total software stack from scratch for every security-demanding application is practically infeasible. After all, deployment of such a solution would totally outweigh the costs of damage done by phishing attacks. The model does not make any statements on the threats related to communication security.

Viral threat model

The Viral threat model can be considered a relaxation of the Thompson threat model [37]. Just like in Thompson's model, the Viral threat model acknowledges malware to be able to penetrate a user's computer and gain control over it. Consequently, the malware is able to control other software that is running on the computer.

However, the main difference with the Thompson threat model is that the Viral threat model assumes the presence of some trusted software. Lifu Wang and Partha Dasgupta proposed secure processors and other hardware solutions to guarantee this [37][89].

Strikingly, the Viral threat model explicitly considers communications insecurity to be a negligible threat. Furthermore, the model makes assumptions

on the behavior of users, merchants and banks. The latter is considered trustworthy whereas the users and merchants are considered to be possibly malicious.

The applicability of this model to a phishing context is questionable. In a two-factor context the trusted software could be in the security calculator. But in a two-channel context it is unclear what the trusted software would be (there is a trusted channel however). Moreover, this threat model is a mix of concerns: It mixes a security model into the threat model by the promotion of security solutions such as trustworthy hardware.

2.4 The phishing threat model

As just discussed none of the previously described threat models can be applied straightforwardly to model phishing in an internet banking context. Therefore, we will derive our own phishing threat model in which the lessons learned from the previously analyzed threat models are merged. For the construction of our threat model we will apply a structured aproach in which we adopt and adapt various modeling techniques which are developed by Microsoft [84].

2.4.1 Identification of internet banking components

We start off by identifying the components that play a role in an internet banking system. We discern the following three categories of components:

- *Processors.* A processor is a computing unit that operates processes (for example a personal computer).
- *Communication channels.* A communication channel is a medium over which data can be exchanged between processors (for example the internet).
- *Actors.* An actor is a human being who operates processors (for example a customer).

Using these categories of components we can model an internet banking system in the Netherlands as follows:

The *customer* (actor) operates a *personal computer* (processor) which communicates with the *webserver of the bank* (processor) over the internet (communication channel). The webserver communicates with the *back-end systems of the bank* (processor) over a *Wide Area Network* (communication channel). Both the webserver and the back-end systems are operated by *bank personnel* (actor). For authentication purposes either a two-factor or a two-channel scheme may be used. In the case of a two-factor scheme a *security calculator* (processor) is used. In the case of a two-channel scheme the customer operates a *mobile phone* (processor) which communicates with the bank's back-end over the *GSM network* (communication channel). Figure 2.1 shows a graphical representation of this model. Note that we added a trust boundary to this picture. This trust boundary demarcates the personnel and computer systems that are under direct control of the bank. The use of this trust boundary will become clear later in this section.

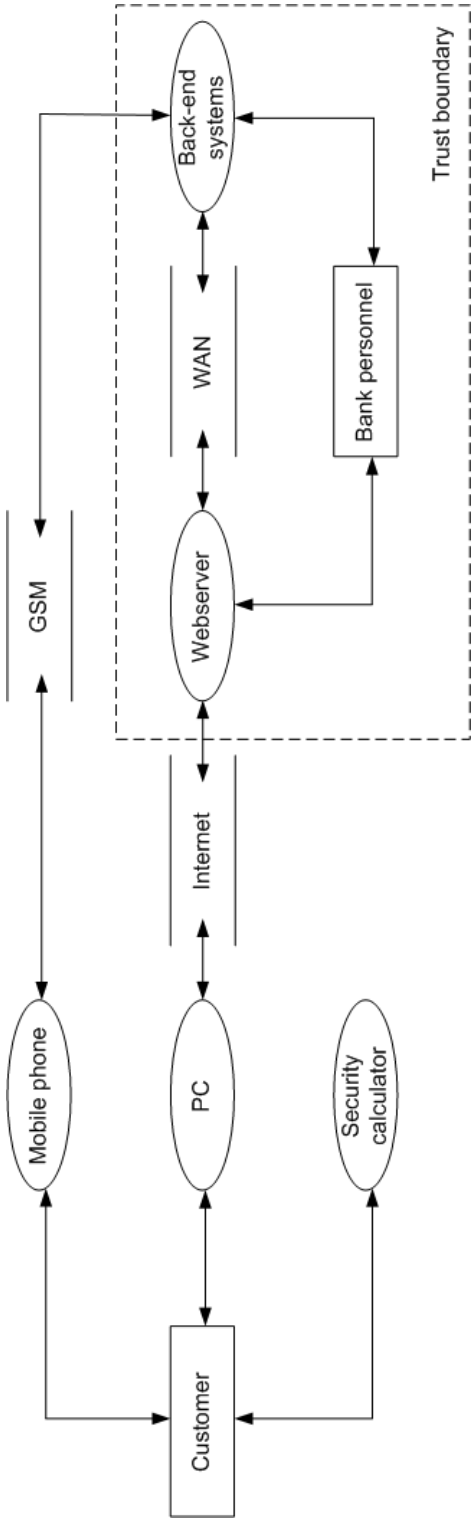

Figure 2.1: Abstract internet banking architecture

Threat	Potentially breaks requirement
Spoofing	Authenticity
Tampering	Integrity, No-intrusion
Repudiation	Non-repudiation
Information disclosure	Confidentiality
Denial of service	-
Elevation of privilege	-

Table 2.1: Relation between threats and requirements

2.4.2 Identification of phishing threats

We will now analyze the threats that the components in our model might face. Two popular branches are employable to carry out such an analysis [67]: Categorized threats list and the STRIDE methodology [52]. The former applies a long list of well-known common threats to the application architecture. However, since specific architectures vary from bank to bank this method does not fit our generic approach. Moreover, the focus on currently known threats renders the application of categorized threat lists impractical for predictions of future phishing attacks.

Instead, we choose to apply the STRIDE methodology which is a more conceptual approach to explore the possible threats. The general idea behind this methodology is that one can group threats into categories according to the STRIDE acronym:

- Spoofing
- Tampering
- Repudiation
- Information disclosure
- Denial of service
- Elevation of privilege

Not all of these threats are relevant in a phishing context. In order to leave the irrelevant threats out of consideration we make a mapping between these threats and the security requirements from Section 2.2.1. This mapping can be found in Table 2.1.
In our phishing threat model we leave repudiation, denial of service and elevation of privilege threats out of consideration. The exclusion of the denial of service threat is obvious since we explicitly ruled out availability from our list of requirements (see Section 2.2). Furthermore, we consider

the threat of elevation of privileges irrelevant in a phishing context since internet banking does not involve complex provisioning models. Lastly, although repudiation directly threatens our non-repudiation internet banking requirement, we leave it out of consideration in our phishing threat model. Repudiation might be a threat aroused by phishing[7] but we do not consider repudiation part of a phishing attack.

This leaves us with the need to analyze the threats of spoofing, tampering and information disclosure for all components in our internet banking model. Note that we added a trust boundary to the model depicted in Figure 2.1. This trust boundary demarcates the personnel and computer systems that are under direct control of the bank. Since banks are under strict supervision and since they employ expert information security personnel we allow for a simplification of our model: We consider the components within the trust boundary to be secure, honest and immune to the threats of spoofing, tampering and information disclosure.

Let us analyze whether the threats of spoofing, tampering and information disclosure are applicable to all types of components in our model. We consider the threats of tampering and information disclosure to be relevant to processors, communication channels and actors[8]. However, the threat of spoofing only applies to information that passes a communication channel. Hence, spoofing is irrelevant to components of the types of processors and actors.

Accordingly, Table 2.2 summarizes the possible threats on the components in an internet banking application (see Section 2.4.1) that may be relevant to phishing:

2.4.3 Phishing threat risk rating

Because of the large number of possible threats we need to prioritize these threats in order to know which threats require focus. Therefore, threats are rated by the risk they represent. For this purpose we define the risk that a threat imposes as the product of the probability that a threat may be exploited and the damage that the exploitation of this threat may cause:

[7] When successful phishing attacks reach media attention, customers might falsely claim to be the victim of these attacks in order to gain compensation.

[8] Tampering with a customer may sound odd. We define this threat as the possibility to delude a customer into diverging from standard operation procedures for internet banking processes.

Component	Spoofing	Tampering	Information disclosure
Customer	-	X	X
Mobile phone	-	X	X
PC	-	X	X
Security calculator	-	X	X
GSM network	X	X	X
Internet	X	X	X

Table 2.2: Mapping between threats and internet banking components

*Risk = Probability of exploitation * Damage potential*
Probability of exploitation and damage potential are rated on an ordinal three-points scale consisting of the values low (1), medium (2) and high (3). This results in a risk scale from one (1) to nine (9) which we will divide into three bands: Low risk (1-2), medium risk (3-5) and high risk (6-9).

Let us evaluate the risk of all possible threats in an internet banking environment that are relevant to phishing:

1. *Customer*

   (a) *Tampering*

Probability	Damage	Risk
High	High	High
Empirical proof of customers' limitation to distinguish genuine from spoofed online material exists [39]. Furthermore, studies show that customers are susceptible to ignoring security indicators and to diverging from correct procedures [81]. Such behavior may be exploited by attackers in highly damaging attack scenarios. Altogether, the associated risk of this threat is large.		

   (b) *Information disclosure*

Probability	Damage	Risk
High	High	High
Studies show computer users easily leak sensitive information [54]. Successful exploitation of this fact may leak data that is critical in an internet banking context such as credentials and TAN codes. Accordingly, the associated risk of this threat is large.		

2. *Mobile phone*

(a) *Tampering*

Probability	Damage	Risk
Low	High	Medium
No examples of serious mobile phone viruses or worms exist. Furthermore, the current heterogeneity of architectures on the mobile phone market renders them a difficult target for large-scale attacks. Although tampering with a customer's mobile phone could result in highly critical attacks (e.g. theft of TAN codes or Man-in-the-Middle attacks) the probability of successful exploitation is so small that the resulting risk is only moderate. Please note that this threat might shift to a high risk when smartphones become more popular since smartphones have security issues very similar to to personal computers [38].		

(b) *Information disclosure*

Probability	Damage	Risk
Low	Medium	Low
A mobile phone may have stored highly sensitive internet banking information such as TAN codes. However, leakage of information from would require physical access to the device. Hence, this threat does not impose a serious risk in a phishing context.		

3. *PC*

(a) *Tampering*

Probability	Damage	Risk
High	High	High
Nowadays the spread of malware is enormous. Visiting a malicious or compromised website is often sufficient to get infected by malware [74]. Furthermore, dubious services exist that spread malware for several cents per infected host [46]. Hence, infecting random internet banking customers can be considered trivial. The damage that successful exploitation of this threat may cause is considerable (e.g. Man-in-the-Middle attacks and stealing credentials and TAN codes). Altogether, the risk of tampering with a customer's PC is extremely serious.		

(b) *Information disclosure*

Probability	Damage Low	Risk
Medium	Low	Low
Information disclosure from a PC involved in internet banking is especially relevant in a roaming scenario or at shared PC's. Credentials are stored on volatile media (i.e. in RAM memory) which might leak in such a context. However, since one-time passwords and TAN codes become invalid after use the value of such information is only slight. Accordingly, this threat imposes a low risk.		

4. *Security calculator*

   (a) *Tampering*

Probability	Damage	Risk
Low	Medium	Low
In the Netherlands all popular security calculators are offline devices. Accordingly, successful tampering would require physical access to the device. Furthermore, since the device is unconnected Man-in-the-Middle attacks are infeasible. Altogether, the threat of tampering with security calculators does not a impose a serious risk in a phishing context.		

   (b) *Information disclosure*

Probability	Damage	Risk
Low	Medium	Low
Since the majority of security calculators in the Netherlands are unconnected information leakage from such devices would require physical access. Moreover, the only valuable information that could leak are digital signatures for transactions that the user initiated or credentials that are valid within a limited timespan only. Consequently, the requirement of physical access combined with the fact that the information that might leak has limited value results in a low risk.		

5. *GSM network*

   (a) *Spoofing*

Probability	Damage	Risk
High	Medium	High
Spoofing SMS text messages is a trivial task for which automated tools are publicly available[9]. Since customers may consider SMS to be a secure medium (after all, banks use it to send sensitive information such as TAN codes) spoofed messages might have serious impact. Altogether, spoofed messages over the GSM network comprise a significant risk.		

(b) *Tampering*

Probability	Damage	Risk
Low	High	Medium
Altough tampering with the GSM network is practically feasible [75] it requires considerable investments from the attacker. Moreover, the attacker needs to be near the handset of the customer. The damage potential is high however: TAN codes might be intercepted. We conclude that tampering with the GSM network is only of medium risk.		

(c) *Information disclosure*

Probability	Damage	Risk
Low	Medium	Low
The attacks for information disclosure of the GSM network roughly require the same costs and skills as tampering with the network while the damage potential is smaller. Consequently, we regard this a low risk threat.		

6. *Internet*

(a) *Spoofing*

Probability	Damage	Risk
High	High	High
Mimicking webpages is trivial and customers seem to easily fall for it [39]. We consider spoofing of digital material over the internet to be a high risk threat. Note that we do not consider IP spoofing to be a feasible attack.		

(b) *Tampering*

Probability	Damage	Risk
Medium	High	High
The DNS mechanism opens up a range of attack vectors [70]. Futhermore, attackers on the customers local LAN or ISP personnel can divert network traffic. We consider tampering with the internet connection of the customer a high risk threat.		

(c) *Information disclosure*

Probability	Damage	Risk
High	High	High
Tools to sniff network traffic are publicly available. Anyone on the network path to the bank (a hacker on the customer's local LAN, the system administrator or ISP personnel) can snoop unprotected traffic, including credentials. Consequently, we consider this a high risk threat.		

Figure 2.2 summarizes the risk that accompanies each of the threats we identified.

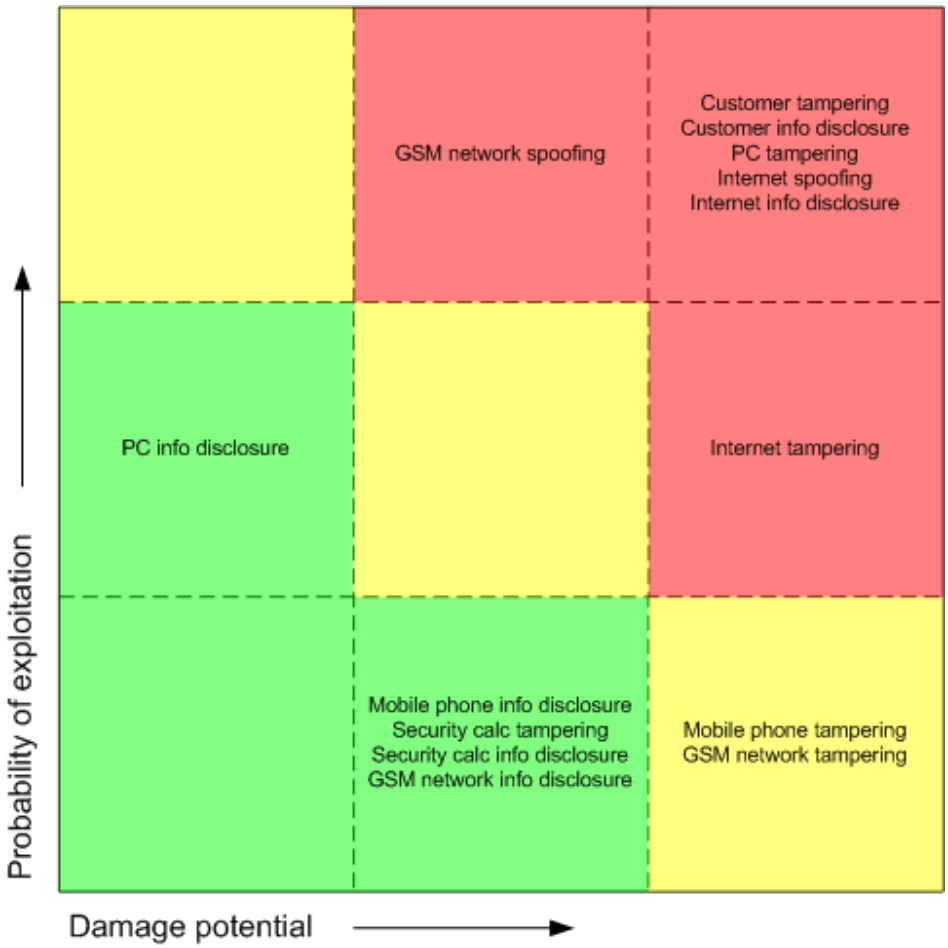

Figure 2.2: Internet banking threats in a phishing context

2.4.4 Adversary model

An essential part of a threat model is an assessment on the resources and capabilities of the attackers. In the Introduction it was explained that phishing gangs are multidisciplinary and well-organized. Consequently, it is sensible and realistic to assume a powerful adversary that is able to simultaneously manipulate all components that are marked vulnerable in the phishing threat model. However, we assume that a phisher does not have physical access to the end-systems of the customers and to the security calculator.

In the next chapter it will be demonstrated how attackers currently exploit

the threats described in our phishing threat model. Later, in Chapter 4 we will analyze how these threats and accompanying attacks are mitigated and to what extent these countermeasures suffice. In Chapter 5 we will exploit the threats described in our phishing threat model to the greatest degree in order to explore which future attacks are feasible.

2.5 Key issues of this chapter

- Recently, media attention for alleged internet banking insecurities seems to have proliferated.
- The public reputation of internet banking is a major issue for Dutch banks.
- Internet banking has become a vital factor in our modern economy. In case of a major security incident (e.g. phishing), for capacity reasons a fall-back arrangement to return to traditional offine banking is not feasible.
- Our security requirements for an internet banking application are: no-intrusion, authenticity, freshness, confidentiality, integrity, non-repudiation and accountability.
- The most relevant non-security requirements for an internet banking application are: roaming, user-friendliness and low-cost.
- Tampering with the customer's end-systems (e.g. personal computer or mobile phone) are threats that accompany a high risk in an internet banking context.
- We assume a phishing gang to be a powerful adversary that is able to simultaneously manipulate all components that are marked vulnerable in the phishing threat model. However, we assume that a phisher does not have physical access to the end-systems of the customers and to the security calculator.

Chapter 3

Analysis of current phishing techniques

In this chapter we analyze the current mode of operation of phishers and their attacks. We devise a systematic overview of how a phishing gang is organized an how their attacks are operated.

3.1 Modus Operandi

This section gives an overview of the current mode of operation of phishers. A systematic approach to classify and identify phishing attacks is presented. Furthermore, a few characteristic examples of real-world attacks will be unraveled.

3.1.1 Actions and actors: A systematic overview

A variety of phishing attacks is carried out in the wild. However, all variants typically share these three core elements: the lure, the hook and the catch[56]. Each of these parts involves its own techniques and proficiencies. Let us categorize and refine these elements systematically by describing a general framework for phishing attacks:

The lure
Goal: Persuade the victim into biting the hook.
Carried out by: Disseminators (see next section for a detailed explanation)
Involves steps:

- *Step 1. Deliver payload*
  The attacker contacts his victims and delivers a payload, for example via an email message or a telephone call. This step regularly involves filter-evasion techniques, such as hiding the message in a GIF image to avoid spam-filters when email is the communication medium to deliver the payload.

- *Step 2. Direct to spoof*
  The payload delivered in Step 1 touches social engineering techniques that direct the victim to the hook. Usually, the payload appears to come from the targeted institution and uses a convincing story that lures the victim to a spoof of this institution.

The hook
Goal: Steal confidential information.
Carried out by: Collectors (see next section for a detailed explanation)
Involves steps:

- *Step 3. Prompt for confidential information*
  Once the victim has navigated to the hook he is prompted with an interface that has the same look and feel as the targeted institution. The deceived is then asked to enter confidential information. Popular information to query are usernames, passwords, credit card numbers, social security numbers, TAN codes and other credentials.

- *Step 4. Leak confidential information*
  The victim might decide to actually enter the confidential information

since he probably feels comfortable with this familiar looking environment.

- *Step 5. Collect stolen information*
  The stolen information is transferred to the attacker. The main goal is to keep the stolen information away from legal authorities since this would render the data worthless. These techniques will be described later in this section.

The catch
Goal: Achieve pay-out
Carried out by: Cashers and mules (see next section for a detailed explanation)
Involves steps:

- *Step 6. Impersonate victim*
  Eventually the attacker will contact a genuine institution. Abusing the stolen information he is capable of impersonating the victim. Some modern phishing attacks perform this step automatically and in real-time parallel to step 4. These and other sophisticated attacks are treated in Chapter 5.

- *Step 7. Achieve pay-out*
  Since the attacker is able to impersonate the victim, the criminal can exploit all services the victim has access to. Usually the pay-out is achieved using obfuscation techniques which will be described later in this section.

Figure 3.1 gives a graphical overview of this systematic approach to phishing attacks.

3.1.2 Organization of a phishing gang: Seperation of concerns

As one can see, a phishing attack is multidisciplinary from the view of the adversary. For a phishing attack to be carried out successfully, an adversary must play different roles. Hence, it is intelligible that most attacks are put into effect by gangs instead of individuals [56]. Traditionally, about a decade ago, full-blown phishing attacks were carried out by individual malicious hackers or at most small groups of a handful of attackers. But meanwhile the scenery of phishing has been taken over by organized crime [20]. As a result phishing attacks have gradually reorganized into a system that follows the principle of separation of concerns[1]. The application of specialists to different aspects of phishing (hook, lure and catch) allows for

[1]For more information see `http://en.wikipedia.org/wiki/Separation_of_concerns`

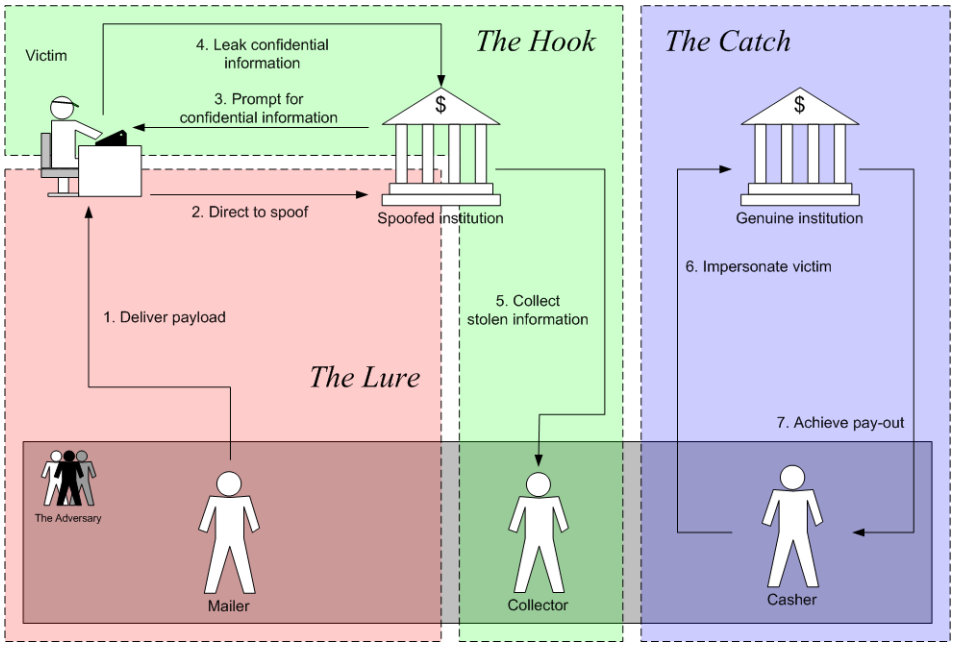

Figure 3.1: Graphical representation of a phishing attack

more sophisticated and effective attacks.

Let us first describe the distinct roles and the organization of the adversary in more detail.

- *Disseminators*
  Task: Disseminators are responsible for delivering the initial payload to the victims.
  Expertise: Their expertise is to evade various filters (e.g. spam filters, anti-virus tools) that attempt to block the payload propagation. Furthermore, disseminators often have the ability to target a large audience, for example through the use of a bot-net.
  Practitioners: Usually this task is performed by malicious hackers or spammers who let their knowledge and tools.

- *Collectors*
  Task: Collectors develop the part of a phishing attack that fraudulently spoofs an institution. Subsequently, they are responsible for gathering the confidential information that is leaked by victims.
  Expertise: The expertise of collectors lies in them being able to create imperceptible forgeries of institution's front-ends. Furthermore, a sly store-and-collect mechanism has to be implemented to collect the stolen information without alerting the targeted institution.

Practitioners: Generally the collectors of a phishing attack are malicious webdevelopers.

- *Cashers*
  Task: A casher is responsible for the exploitation of the stolen information. In order to do so, he contacts a legitimate institution and impersonates a victim by presenting the stolen information. Thereupon, an account or service the victim has access to is abused for the gains of the casher.
  Expertise: Financial fraud. Additionally, cashers must know how to cover their tracks by using anonymity enhancing techniques such as the use of proxies or the Tor network.
  Practitioners: Often members of old-fashioned organized crime, significantly less technically oriented than disseminators or collectors.

- *Mules*
  Task: The catch of a phishing attack is generally a collection of several money laundering techniques. A major pivot in this part is the so-called mule. Mules are intended as low-level couriers in the criminal organization of phishing. Their job is to transfer money that they received on their own account to another account controlled by the attacker. They thereby obfuscate the money flow, which makes it significantly more difficult for targeted institutions to trace the actual attackers and to reclaim the losses.
  Expertise: Barely nothing. Anyone who can make a money transaction can become a mule.
  Practitioners: Usually naïve, innocent victims. Very often, mules are recruited using spam and fake vacancies sites. The attacker offers a position like "Transfer Manager". Once a candidate is found, his job is to transfer the money (minus a small percentage as fee) that he received to another account. The lifecycle of a mule is short: He is easily targeted by the damaged institution and subsequently the mule awaits prosecution.

The phishing supply chain

A common pattern for the organization of a phishing attack is where the propagation of the phishing lure is outsourced to an external party. It is supposed to be very uncommon for a disseminator and casher to be the same individual since these functions require different expertise. To support this organization a well-structured supply chain has developed. Public Internet Relay Chat channels exist which act as a black market place for phishers. One of these channels has been extensively researched by academic researchers from January to Augusts 2006 [46]. During these periods these researchers identified thousands of offerings and wanted services that apply

to all stages of a phishing attack. For example, about 20% of the advertisements concerned offerings or requests for phishing lure services, such as disseminators, hacked hosts and email address lists. About 5% of the advertisements offered stolen internet banking credentials. Furthermore, several services for money laundering were offered on the channel. This vision on the phishing supply chain is shared by RSA researchers [20].

3.1.3 Phishing techniques

Phishers utilize a number of technical tricks to improve the efficiency and effectiveness of their attacks. Below is a description of prevalent techniques, fit into our phishing framework.

Techniques to improve the lure

- *Step 1. Payload delivery*
  Spamfilter evasion. Nowadays, many Internet Service Providers, corporations and home users apply email filtering technology to prevent unsolicited email messages from arriving in users' inboxes. These filters are capable of successfully filtering out more than 95% of all undesirable messages [12]. Obviously, attackers seek for ways to get around these filters in order to efficiently propagate their phishing payload. Various tricks exist to achieve this goal. For example, attackers can embed their message in a picture to prevent filters from analyzing the text of the message. Furthermore, HTML email opens a range of opportunities for attackers. A popular technique is to include random text in the message that has the same color as the message's background. Consequently, a naïve filter will take this text into account upon examination of the email and may be fooled by it. Moreover, the payload is not affected since an average user will not spot this invisible text.

  Bot-nets. A bot-net is a network of a large number of PC's that are infected with Trojan horse software. Consequently, the keeper of the network has control over all contaminated nodes without the consciousness of their legitimate users. A bot-net is an ideal platform to dispatch a large number of messages since it allows for a distributed message propagation system. There are many sources the messages come from and thus blacklisting originators of messages is impracticable. Additionally, tracing back a message ends at the PC of a user who is ignorant of the crime and hence leaves little clue into the direction of the actual criminal.

  Drive-by installation. Installation of malware code can be done us-

ing a technique called drive-by installation. In drive-by installations cracked websites or online advertizing spaces are used to publish malicious HTML code. This HTML code exploits a vulnerability (e.g. a buffer overflow) in a popular webbrowser. When a customer happens to browse to such a site the malware is automatically installed on his computer. For research on drive-by installations we refer the reader to [73].

- *Step 2. Direct to spoof*
  Email spoofing. The Extended Simple Mail Transfer Protocol is the de facto standard for email transmission across the internet [60]. One of the aspects of this protocol is that it has no facility for the authentication of senders. Consequently, the originator address of a message can be easily spoofed. As a result, email spoofing is a common technique used by disseminators to let a message appear to come from a genuine institution.

  Personalization. By obtaining information about a victim such as full name, date of birth, type of job or the names of social contacts the lure can be personalized. Adopting this information an attacker can make the lure look more reliable and authentic. Personal information is increasingly available on the World Wide Web since the birth of community websites such as LinkedIn[2], Hyves[3] and MySpace[4]. Personalization techniques are an essential ingredient of spear-phishing attacks, an emerging variant of phishing which is described in Section 5.2.1.

  URL hiding. A popular technique to direct a victim to the hook is URL hiding [70]. The HTML payload contains a hyperlink that appears to lead to a legitimate website but when clicked the browser navigates to a site controlled by the attacker. A basic trick is by supplying an anchor tag with a deceptive link description:

```
<a href="http://www.attacker.com">
http://mijn.postbank.nl</a>
```

  However, this trick is easily detectable since the actual direction of the link will appear in the lower left corner of the browser as soon as the user moves his mouse over the link. A more advanced deception can be achieved by incorporating Javascript:

[2] `http://www.linkedin.com`
[3] `http://www.hyves.nl`
[4] `http://www.myspace.com`

```
<a href="http://mijn.postbank.nl"
onClick="document.location='http://www.attacker.com';
return false;">Internetbankieren</a>
```

Social engineering. Frequently, social engineering techniques are of major importance in the lure of phishing attacks. A convincing reason is required to persuade a user to follow the path to the hook. For example, the victim is sent an email that describes a transaction initiated from his account that he obviously did not make. Possibly, the user clicks the link that is included in the message that promises the possibility to cancel the transaction. Subsequently, he is taken to the hook where he is prompted to enter his credentials. Another example of social engineering used in the lure is to promise a financial incentive to the victim. In order to receive the price or discount the victim is lured to the hook where he has to enter his account details.

Techniques to improve the hook

A general technology used to improve the hook of a phishing attack is the application of a so-called toolkit. Such a toolkit combines several of the techniques described here. An example of a phishing toolkit is treated later in this section.

- *Step 3. Prompt for confidential information*
  Techniques used in this step generally exploit human-computer interaction aspects. Several of these techniques and descriptions of why they work are included in more detail in Section 4.1.2. Here follows a short impression:

  Resembling URL's. Attackers regularly host their spoofs at hostnames that resemble the ones of the targeted institution. For example, attackers may use a domain name such as postbank-secure.com to mimic a legitimate Postbank domain.

  Browser spoofing. Technologies such as client-side scripting (e.g. Javascript), HTML and Cascading Style Sheets are powerful tools to mimic the user interface and behavior of a web browser. Exploiting these technologies attackers are capable of concealing or spoofing security indicators like the SSL padlock.

- *Step 4. Leak confidential information*
  At the current state of phishing attacks the technological sophistication of this step is very limited. However, this is one of the aspects

in phishing where I expect some major technological improvements in the near future. This topic is elaborated in Section 5.3.

- *Step 5. Collect stolen information*
  Two mechanisms are in common use for this:

  Batch-mode. The hook temporarily stores the gathered information until the attacker comes to collect the data. From an attacker's point of view this procedure has the advantage that the attacker can pick up the stolen information whenever he wants from where he wants. A drawback is that the attacker has to install a protection mechanism to prevent system administrators or prosecutors from obtaining a stored data batch.

  Singleton-mode. Every record of stolen information is immediately forwarded to the attacker (for example via email). From the viewpont of the attacker this has the advantage that the booty is separated from the spoof. Consequently, if legal authorities cease the spoof the stolen information is maintained and remains valuable since the authorities have no insight in which information was stolen.

Techniques to improve the catch

The main challenge for the attacker while performing the catch is not to be caught himself. Attackers utilize some technological mechanisms to accomplish this goal.

- *Step 6. Impersonate user*
  Privacy enhancing technologies. Privacy enhancing technologies are technologies that allow for anonymous use of internet services (to a certain extent). They can serve the needs of honest internet users who, for example, are oppressed by dictatorship or are involved in whistle-blowing. However, on the other side privacy enhancing technologies are also suitable for supporting malicious activities. Cashers make use of technologies such as Tor [41] and anonymous proxies to cover their tracks when logging into WWW services using stolen credentials.

- *Step 7. Achieve pay-out*
  Money laundering. In order to achieve a pay-out several money-laundering techniques are used. A popular manner to exploit credit card information is to purchase valuable goods and sell these on the black market. Internet banking credentials are usually used to create a tangle of payments using mules after which the money is transferred to a foreign account.

3.2 Popular variants of phishing

Many realizations of the aformentioned concepts are carried out. Let us describe some variants of phishing that are most prevalent. More sophisticated and about-to-come variations are treated in Chapter 5.

Last year only two classes of successful phishing attacks on Dutch internet banking services reached publicity. We entitle these attacks real-time man-in-the-middle phishing and malware-based phishing. A third form of phishing, entitled dragnet phishing, is not very successful in the Netherlands but is extremely popular in other countries[5]. We will now discuss these attack classes in more detail.

3.2.1 Dragnet phishing

Worldwide one of the most common forms of phishing is the so-called dragnet phishing. In this approach the lure consists of a mass-mailing to thousands of potential victims. Their email addresses are usually bought from criminals who run webspiders to harvest email addresses from the World Wide Web. Consequently, the receivers of the email might not even have an account at the target institution, which considerably mitigates the success ratio of the attack. The mass-mailing is done with the use of a botnet, which is a network of compromised machines.

The hook of dragnet phishing consists of an imitation of the website of a legitimate institution. Such a spoof closely resembles the look and feel of the authentic site. The spoof asks the victim to enter confidential information, which is subsequently sent to the attacker.

Note that dragnet phishing attacks mainly target trivial password-based authentication schemes. In the Netherlands more advanced authentication schemes are employed (see Section 4.1) that render these attacks infeasible.

Example of dragnet phishing: the Rock-phish attack

Rock-phish attacks aroused by the end of 2005 and are held responsible for about half of the number of reported phishing incidents worldwide [68]. The lure of a rock-phish attack is formed of a bulk mail which evades spam filters by incorporating some random text and a GIF image which contains the actual message. Moreover, Rock-phish attacks make use of a toolkit in order to obtain a sophisticated and distributed hook. The attackers maintain a

[5] This class of attacks mainly targets trivial password-based authentication schemes. In the Netherlands banks employ two-factor and two-channel authentication schemes which render these attacks infeasible as we will discuss in Chapter 4.

network of compromised hosts where each host is loaded with several fake bank websites. The toolkit makes it possible to operate all these spoofs from a single domain name.

Let us discuss the steps carried out in such an attack.

The Lure

- *Payload delivery*
  Using a bot-net a bulk mail is sent to a large number of customers. This email evades spam filters by incorporating some random text and a GIF image which contains the actual message.

- *Direct to spoof*
  The GIF image contains a social engineering text that deceives the customer to visit a specific URL. At this URL a spoofed internet banking website is located.

The Hook

- *Prompt for confidential information*
  The spoofed internet banking website asks the customer to login using his credentials.

- *Leak confidential information*
  If the customer believes the spoof is a genuine internet banking website he might decide to enter his credentials. The spoof immediately stores this information and displays an error message or redirects the user to the genuine internet banking website.

- *Collect confidential information*
  As just discussed the Rock-phish attack employed a toolkit that contains spoofs for many internet banking websites. Accordingly, the toolkit stores credentials for many internet banking applications. Hence, the attackers can collect stolen credentials in batch mode.

The Catch

- *Impersonate user*
  The attackers use the stolen credentials to login to the corresponding internet banking applications and to initiate malicious transactions.

- *Achieve pay-out*
  Unfortunately, no details of a money laundering scheme employed by these attackers is publicly available.

3.2.2 Real-time man-in-the-middle phishing

Real-time man-in-the-middle phishing is a class of phishing attacks that caught much media attention last year. These attacks demonstrated weaknesses in two-factor authentication schemes (see Chapter 4 and Chapter 5 for a deeper discussion on these weaknesses).

In real-time man-in-the-middle phishing a spoofed website that mimics the attacked internet bank is employed. This spoof can communicate in real-time with the genuine internet banking website. As a consequence, the spoof can present the true look and feel of the online internet banking application.

Example of real-time man-in-the-middle phishing: ABN Amro

One of the attacks that caught most media attention was a man-in-the-middle attack on the internet banking services of ABN Amro in March 2007. At least four customers are known to have been compensated for unknown amounts that were stolen from their accounts [6].

This infamous attack is especially known for its demonstration of weaknesses in two-factor authentication systems. ABN Amro implements an authentication scheme in which an external device, called a security calculator, is used. This calculator is used in combination with a challenge-response protocol in order to generate tokens that can be used to log in to the system and to initiate transactions. Let us fit this attack into our phishing framework.

The Lure

- *Payload delivery.*
  The attackers sent a mass-mailing falsely claiming to be from ABN Amro. Once customers opened the malicious attachment in the email malware was installed on their computers.

- *Direct to spoof.*
  As the malware took control over their computers, as soon as customers typed in the URL of the ABN Amro internet banking website in their web browsers they were silently redirected to a website hook.

The Hook

- *Prompt for confidential information.*
  The website to which customers were redirected completely copied the look and feel of the genuine ABN Amro internet banking website (except for the SSL padlock).

[6] See `http://www.theregister.co.uk/2007/04/19/phishing_evades_two-factor_authentication/` for more information.

- *Leak confidential information.*
  Since customers thought they were dealing with a genuine ABN Amro website they entered their credentials and initiated transactions. Note that because of the two-factor authentication system implemented by ABN Amro these credentials were only valid for a single use.

- *Collect confidential information.*
  The website hook saved all information entered by the customers and communicated with the genuine ABN Amro internet banking website to communicate all information required for the challenge-response protocol that the two-factor security calculator implemented.

The Catch

- *Impersonate user.*
  The impersonation happened in real-time with the actions of the customers. As soon as the customers initiated a transaction this transaction was blocked at the website spoof. Instead, the captured tokens created by the security calculator were used to initiate malicious transactions.

- *Achieve pay-out.*
  Comprehensibly, ABN Amro did not disclose details on the pay-out achievement tricks used by the attackers. But strikingly, immediately after the attack was noticed the bank removed the Urgent payment options from their internet banking systems. So it seams reasonable to assume that this option was used by the attackers to initiate quick transactions on which the ABN Amro bank could not perform chargebacks.

3.2.3 Malware-based phishing

In the last few years malware has been incorporated into phishing attacks [19]. These malware-based attacks do not use spoofed websites to steal confidential information but they use malicious pieces of software that are to be installed on the customer's end-systems (e.g. personal computer).

Example of malware-based phishing: Postbank TAN-code trojan

In March 2006 a piece of malware was found that targeted the Dutch bank Postbank. Let us look at this attack in more detail.

The Lure

- *Payload delivery*
  Installation of the malware was done using the aforementioned technique of drive-by installations.

- *Direct to spoof*
  The malware hooked web browser functionality (see Section 5.3.1 for an example of how to do this). Thereby it was able to detect when the customer visited Postbank's internet banking website.

The Hook

- *Prompt for confidential information*
  The malware simply awaited genuine internet banking behavior of the customer.

- *Leak confidential information*
  As soon as the customer entered login credentials or TAN codes the trojan recorded this information. Moreover, when the trojan encountered that a TAN code was entered it used the hooked web browser to cancel the transmission of the TAN code and display an error message.

- *Collect confidential information*
  All valuable information (login credentials and TAN codes) were sent to the attackers in singleton mode over the internet.

The Catch

- *Impersonate user*
  The attackers used the credentials to login to the internet banking application of Postbank. Subsequently, a transaction was created using the stolen TAN code.

- *Achieve pay-out*
  Unfortunately, no details of a money laundering scheme employed by these attackers is publicly available.

3.3 Reflection on threats

To conclude this chapter we will analyze which threats from our threat model (see Section 2.4) are actively being exploited by the phishing attacks described in this chapter. In this way we can identify which threats should be mitigated to successfully counter current phishing attacks. In order to do so we will recall the threats identified in Section 2.4 and indicate whether they are currently being exploited. For threats that are being exploited we will point out an attack by which the threat is typically being exploited (either dragnet phishing, Man-in-the-Middle phishing or malware based phishing). This can be found in Table 3.1.

It is not a surprise that the threats that are actively being exploited by phishers are exactly the ones that we assigned a high risk rating in Section 2.4. However, it is a surprise that phishers succeed in exploiting such a large number of these threats (5 out of 7 high-risk threats are exploited). This suggests that these threats are insufficiently mitigated by defensive controls. In the next chapter we will discuss this issue extensively.

Threat	Risk	Exploited	Example attack
Customer tampering	High	Yes	Dragnet phishing
Customer information disclosure	High	Yes	Dragnet phishing
Mobile phone tampering	Medium	No	-
Mobile phone information disclosure	Low	No	-
PC tampering	High	Yes	Malware based phishing
PC information disclosure	Medium	No	-
Security calculator tampering	Low	No	-
Security calculator information disclosure	Low	No	-
GSM network spoofing	High	No	-
GSM network tampering	Medium	No	-
GSM network information disclosure	Low	No	-
Internet spoofing	High	Yes	Dragnet phishing, MitM phishing
Internet tampering	High	Yes	MitM phishing
Internet information disclosure	High	No	-

Table 3.1: Exploitation of internet banking threats by phishing attacks

3.4 Key issues of this chapter

- Phishing attacks can be modeled into three parts: the lure, the hook and the catch. Each of these parts can be refined into further steps.
- In a phishing gang we can identify a separation of concerns that enables multiple roles. Namely the roles of disseminators, collectors, mules and cashers.
- The separation of concerns leads to a phishing supply chain which is supported by illicit digital trading places.
- Dragnet phishing, real-time man-in-the-middle phishing and malware based phishing are the most prevalent forms of phishing.

Chapter 4

Analysis of current phishing defenses

A huge variety of phishing defenses are available. These solutions range from malware scanners to complex transaction anomaly systems installed at banks. In order to discuss this wide range of defenses orderly we introduce a taxonomy that categorizes phishing defenses into front-end and back-end security solutions. The former are solutions that customers deal with directly, whereas the latter are phishing defenses that are installed at the site of the bank. For both categories we analyze solutions that are popular in a Dutch internet banking context. Finally, we combine the front-end and back-end solutions and examine the security achievements of the defensive strategy as a whole.

We do not aim to give a complete overview of available anti-phishing controls. We only thoroughly treat anti-phishing controls that are most prevalent in the Netherlands. For other defensive mechanisms we refer the reader to other studies. For example, e-mail authentication [24][24], virtualized browsers [27], e-mail filters [43], extended validation certificates [3][53], anti-phishing toolbars [35][79] and enhanced authentication protocols [80][48][86] are not discussed in this section.

4.1 Front-end security solutions

Front-end security solutions are anti-phishing measures that involve internet banking customers directly. In the Netherlands two classes of such measures are extremely popular. Namely, end-system security products and authentication mechanisms. In the following sections these anti-phishing solutions will be extensively discussed. Please note that we focus on technological solutions. Hence, solutions that target user behavior are only marginally discussed.

4.1.1 End-system security products

End-system security products are in place to enhance the software security of the personal computer of the customer. Accordingly, installation and maintenance of these products are the sole responsibility of the customer. Solutions that are by far the most popular in this category are malware scanners and personal firewalls. A significant property of end-system security products is that the end-user is fully responsible for their installation, maintenance and correct usage. Note that this might have serious consequences for our roaming requirement (see Section 2.2.2).

With respect to the security requirements for internet banking (see Section 2.2.1) these solutions respectively serve integrity and confidentiality goals. The 3xKloppen campaign[1] heavily promotes the need for these end-system security products in order to guarantee internet banking security. In the comming sections we will analyze whether this claim is reasonable and we will discuss to what extent end-system security products establish or break internet banking requirements.

Malware scanners

Malware scanners endeavor to keep malicious software from end-systems. In this way an attempt is made to create a secure environment in which the software that is required to make use of internet banking services (e.g. the web browser, keyboard drivers and network drivers) can operate. Consequently, any information entered by the user should be submitted unimpaired by the software on the personal comuter.

These scanners are available in two variants: On-access and on-demand scanners. The first are scanners that are always active in the background whereas the latter start scanning activities upon request by the user. Popular on-access scanners are Symantec AntiVirus and Kaspersky AntiVirus.

[1] The 3xKloppen campaign is a campaign by Dutch banks in order to raise internet banking security awareness of customers. See `http://www.3xKloppen.nl` for more information.

Well-known examples of on-demand scanners are Lavasoft's AdAware and Spybot Search & Destroy.

The deployment scale of virus and malware scanners is extremely large. 91% Of Dutch internet users claim to have a virus scanner installed [17]. Accordingly, this widespread propagation creates serious constraints for phishers as they have to evade these scanners. Unfortunately, malware scanners have some major issues that facilitate the development of malicious software that has capabilities to evade these scanners.

The cause of these issues is the manner in which malware scanners detect maliciousness. The majority of virus scanners apply fingerprinting techniques in order to detect malware. This technique involves scanning the binary code for known malicious patterns. The major benefit of this techniques is that it is accurate and hence only yields a small number of false positives.

However, this method also has considerable deficiencies. As a consequence of this detection strategy virus scanners have dreadful response times against phishing attacks that incorporate malware. First of all a piece of malware has to be detected in the wild or using a honeypot [72]. Subsequently, a signature to detect the piece of malware has to be developed. Then, a bunch of signatures has to be collected to release an update batch. Finally, an instance of the scanner will probably look for an update on not more often than a daily basis. Altogether, this detection process may take several days or even weeks [14]. The malware attack on ABN Amro in March 2007 demonstrated this lack of timely response perfectly[2]. Another technology that demonstrates this fundamental problem are polymorphic viruses that change their appearance before the anti-virus industry is able to roll out a signature [91].

Because of this lack of a quick response anti-virus vendors are caught in an endless cyclic battle with malware developers. Since the detection process of a new piece of malware takes several days to weeks the malware developers have a substantial amount of time available to morph their malware and thus reset the detection process [32]. In this manner, the anti-virus industry structurally lags behind the attackers.

Another issue with malware scanners is the requirement of user-friendliness (see Section 2.2.2). Installation, maintenance and correct interpretation of

[2]Empirical evidence are the response times of popular anti-malware products against the ABN Amro phish at `http://www.security.nl/article/15722/1/ABN_Amro_klanten_doelwit_van_malware_aanval_\%2Aupdate\%2A.html`

scan results from these products may require computer skills that go beyond that of non-technical computer users such as elederly. Moreover, many malware scanners are costly. For example, Norton Antivirus 2008 costs over 50 euros for a one-year license[3].

All in all, malware scanners are a necessary but not a sufficient condition to protect against phishing attacks. These scanners should absolutely not be considered a silver bullet against phishing, especially because of the large response times which render them obsolete against swift malware spread. Nevertheless, these scanners limit the time-span and scale on which phishers can operate. Malware scanners force phishers to act timely or to develop stealthy tailored malware, which requires larger investments.

Personal firewalls

Personal firewalls create an additional layer of security with respect to phishing by imposing control on network traffic. These firewall products can defend against malware-based phishing attacks, such as attacks carried out using trojans or keyloggers. In the case of such an attack any captured information needs to be transmitted over the internet to the phisher. Personal firewalls can prevent or at least complicate this transmission process. Popular personal firewall products are ZoneAlarm and Microsoft Windows Firewall.

The usage of personal firewall products is widespread: 78% Of Dutch internet users claim to have a firewall product installed [17]. Although this propagation is not as much as that of malware scanners, it still imposes serious restrictions on the capabilities of malware. A straight-forward socket connection from the malware process to transmit stolen credentials has a high probability of being blocked. This requires phishers to invent firewall-evading tricks, such as process injection or using the Internet Explorer-DOM to transmit stolen credentials, which enlarges a phisher's required investments.

The costs of personal firewalls are similar to those of virus scanners. Commercial products are available at costs up to several dozens of dollars annually (for example, ZoneAlarm Internet Security Suite costs $50[4]). Installation and maintenance efforts are also similar to those of virus scanners.

But as opposed to malware scanners, many personal firewall products also require extensive configuration effort. A study shows that the prompts that

[3]Retail price on bol.com by January 2008

[4]Which is the official retail price by January 2008, see `http://www.zonealarm.com/store/content/home.jsp`

accompany this configuration process confuse non-technical users [36]. Consequently, personal firewall products seem to infringe our user-friendliness requirement.

Moreover, personal firewalls suffer from a chicken-and-egg problem. In a phishing context the task of a personal firewall is to protect against malicious software that leaks credentials. However, since this malicious software is installed on the same system as the personal firewall this enables the malware to turn off or at least reduce firewall functionality.

4.1.2 Authentication mechanisms

Authentication mechanisms can be grouped into two categories: server authentication mechanisms and customer authentication mechanisms. The former are in place to prove the identity of the bank's assets, in particular the internet banking website, to the customer. In a phishing context, these mechanisms are valuable in fighting the hook of a phishing attack. SSL/TLS is by far the most popular mechanism for server authentication. Besides, customer authentication mechanisms aid in validating the identity of the customer when he performs internet banking operations (e.g. establishing an internet banking session or initiating transactions). These mechanisms are a counter measure against the catch of a phishing attack, especially with respect to the impersonation step.

The weaknesses of password authentication

Many foreign internet banking services still rely on a trivial password protection. In such a scheme, in order for customer Alice (A) to establish an internet banking session with the bank (B) she supplies her identity and her password P:

(1) $A \longrightarrow B : A, P$

However, this scheme is hopelessly vulnerable to a phishing attack. A phisher Eve (E) could lure Alice to a spoofed website that mimics that of B and perform a Man-in-the-Middle attack. When Alice leaks her password to the spoof Eve could replay Alice's credentials and successfully initiate an internet banking session by impersonating Alice:

(1) $A \longrightarrow E : A, P$
(2) $E \longrightarrow B : A, P$

Message (1) of this attack is part of the hook of a phishing attack and successfully violates the server authenticity requirement we derived in Section 2.2.1: Alice lacks a suitable mechanism to verify the genuineness of the internet banking website and as a result of this she leaks her credentials to a spoofed website. Message (2) is part of the catch of this phishing attack and demonstrates that our freshness requirement (see Section 2.2.1) is not met by this scheme: The impersonation step (as described in Section 3.1) can be carried out just by replaying the stolen credentials.

As just demonstrated, a trivial username-password authentication scheme is inadequate to establish our security requirements. Dutch banks target this problem in two ways. The authenticity problem in the hook of the previously

described attack is counteracted by the deployment of SSL/TLS. The impersonation act in the catch of this attack is targeted by the deployment of an advanced customer authentication scheme. Almost all Dutch banks incorporate a two-factor authentication mechanism to authenticate their customers. Postbank is the only major Dutch bank that uses a two-channel variant to authenticate their customers. Both the server authentication mechanism using SSL/TLS and the customer authentication schemes using two-factor and two-channel techniques will be treated extensively in the upcoming sections.

Server authentication and link encryption with SSL / TLS

In order to ensure the authenticity of an internet banking website and to provide private and reliable communications virtually every online internet bank supports SSL/TLS. Secure Sockets Layer (SSL) was originally developed by Netscape. In 1996, version 3.0 was released which later served as the basis for the Internet Engineering Task Force standard protocol Transport Layer Security [40].

SSL / TLS functionality

These protocols provide a secure connection between the customer's browser and the webserver of the internet banking service. The connection is secure in the sense that it is resistant against both passive (eavesdropping) and active (packet injection and modification) network attacks. Confidentiality of the connection is ensured by using symmetric encryption for which a session key is negotiated. Furthermore, the integrity of the connection is established by a message integrity check that assures that information that passes the connection has not been altered in transit.

SSL/TLS supports verification of the identities of both ends of the connection by cryptographical signatures. Certificates issued by a trusted third party are supported to ensure the validity of these signatures. Note that in most cases this authentication is only one way proving the identity of the bank to the customer. In currently popular setups SSL / TLS does not authenticate the user to the bank. Accordingly, additional authentication mechanisms to authenticate the user are required (see the upcoming sections on Customer authentication mechanisms).

In practice, the trusted third parties that issue certificates to ensure signature validity are the so-called Certificate Authorities (CA's), for example Verisign and GeoTrust. These CA's are organized in a hierarchy of trust in which a CA that is higher in the hierarchy can issue certificates that guarantee the validity of certificates issued by a CA that has a lower rank. The browser of the customer has the root certificates that cover the most

common CA's preinstalled. Accordingly, it is up to the user to verify the security indicators in the web browser in order to be ensured that indeed a SSL/TLS secured connection to a legitimate banking website has been setup. Verification of these security indicators comprehend two steps: (1) verifying the validity of the URL[5] and (2) checking the presence of the SSL padlock icon in the browser chrome.

Weaknesses of SSL/TLS implementation

Unfortunately, it is with these security indicators where things seem to go wrong. Studies demonstrate that users have a tendency to ignore these security indicators. In a study performed by researchers from Harvard University and UC Berkeley answers to the question what makes a bogus website credible are found. In this study Rachna Dhamija et al. found that well-constructed phishing websites fooled 90% of the participants [39]. The main technological factor in this is the ease of visual deception. Common web browser technologies such as DHTML and AJAX allow web developers to create fancy websites that present advanced user interfaces. The drawback is that this technology can also be used malicious entities to create sophisticated spoofs. In a lab experiment researchers demonstrated the ease of mimicking all browser security indicators [90]. Using Javascript and DHTML the researchers were capable of spoofing the SSL padlock, SSL Certificate Information, the status bar and even address bar interaction.

Another outcome of the study of Dhamija et al. is that the majority of participants of their study could not correctly explain the meaning of the web browser's SSL padlock and the meaning of SSL certificates. Moreover, these researchers identified the problem that users do not understand which parts of the browser are controlled by the visited website and which are not. Users were easily fooled by a SSL padlock placed in the contents of a website or a SSL padlock *favicon* instead of one which is placed in the *browser chrome*[6].

This problem is magnified by the customer's lack of knowledge of the domain name system which complicates the verification of the URL of the internet banking service. Phishers actively exploit this by registering top-level domain names that mimic authenticity. For instance, phishers registered the domain name `hsbc-onlinebankings.co.uk` for phishing pur-

[5]Which introduces the problem that a customer should now what the correct URL looks like as will be discussed later in this section.

[6]The borders of a web browser window, which include the window frames, menus, toolbars and scroll bars.

poses that target the HSBC bank.[7] Unfortunately, instead of mitigating the domain name recognition problem banking procedures often only amplify it. Many banks practice the habit of hosting websites at various domains (e.g. secure-bank.nl) instead of creating subdomains (e.g. secure.bank.nl). For example, Bank of America hosts a web site at the third party reo.com domain: `http://bankofamerica.reo.com`. Another example is `http://accountonline.com` which is a Citibank domain. Such habits confuse customers and phishers actively exploit this aspect. Fortunately, some major Dutch banks seem to be aware of this issue. Interviews with experts demonstrated that at some banks policies are in place to host all internet activities as a subdomain under the primary domain name.

Moreover, implementation issues severely limit the functionality of the SSL/TLS system. Most importantly, the monthly SecuritySpace survey shows that there is a problem with the majority of SSL/TLS certificates such as certificates being self-signed, expired, etc. In November 2007 68% of all certificates encountered were invalid [21][8]. Upon visiting a site that has such a certificate installed the web browser will popup a warning message. Consequently, the large number of invalid certificates supports users to develop a habit to ignore warning messages related to SSL certificates. This problem was sharply formulated by computer scientist Peter Gutmann from the University of Auckland:

"An entire generation's computing experience is built around clicking `OK` to error messages that they do not understand." [50]

Computer use is inherent to regularly perceiving cryptic error messages. This has led to user conditioning to click away error messages. Empirical evidence supports this: Popup warnings about fraudulent certificates are ineffective. About 70% of the participants in the study of Dhamija et al. proceeded without hestitation when presented with warnings.

These problems related to SSL/TLS and security indicators are amplified by CA's and bank's habits. Many Certification Authorities and banks provide logos that promise verified security. Examples of such logos are depicted in Figure 4.1. These logos are easily mimicked or copied, give a false perception of security and increase the confusion that surrounds valid security indicators.

[7] For more details on this particular attack see `http://www.millersmiles.co.uk/report/5893`

[8] Altough we doubt the reliability of this extremely high number we think it is a right assumption that customers regularly encounter self-signed and even expired certificates.

Figure 4.1: Examples of logos that may give a misleading sense of security

This false perception of security is also the symptom of another problem. Some users seem to interpret websites analogous to the real world. In the real world judgements of authenticity are based upon objects such as an impressive storefront. An example is the assessment of authenticity based upon a flash animation "because that would take a lot of effort to copy" [39]. However, in the digital world an impressive storefront is easy to copy. But how should the majority of WWW users who do not have web development experience know this?

Two-factor customer authentication

Authentication can be based upon any of the following factors:

- *What you know.* For example authentication based on passwords.
- *What you have.* For example authentication based on the possession of a token.
- *Who you are.* For example authentication based on fingerprints.
- *Where you are.* For example authentication based on IP address information.

An authentication scheme that is based on a combination of these factors is called multifactor authentication. In a Dutch internet banking context a combination of the two factors of *What you know* (PIN code) and *What you have* (banking card) is extremely popular.
Almost all Dutch banks employ this two-factor authentication scheme to authenticate their customers. In such a scheme all internet banking customers

own a small and portable computing device which we will call a *security calculator* in the rest of this section. For example, Rabobank has rolled out a security calculator under the name of *Random Reader*. Likewise, ABN Amro uses a similar device called *e.dentifier*. VASCO is a major world wide producer of these security calculators, especially its security calculators in the Digipass 800 product range[9] are popular among Dutch banks. These security calculators are used to assist the authentication process and the integrity of all important actions during a internet banking session, such as logging in, sending a transaction request and to transfer money from a savings account.

The security operations of these calculators are based on a shared secret between the bank and the security calculator. In currently popular systems, derivation of this shared secret is based upon the entry of the smart card (in which the secret is stored) and the correct PIN code. Consequently, this allows for a system in which all security calculators are identical, which is convenient from the perspective of both the bank and the customer. Namely, replacement of a broken device can be done readily. Moreover, since these calculators are interchangeable any customer can borrow a security calculator from a friend or a colleague who uses internet banking services from the same bank. This interchangeability feature is in favor of our roaming requirement (see Section 2.2.2). Legacy systems used a security calculator which held the shared secret itself. Consequently, roaming was more difficult and the production costs of these unique devices was higher.

The security calculator is responsible for achieving two goals:

- Establish a verifiable binding between the identity of the customer and the logon action at the beginning of an internet banking session.
- Establish a verifiable binding between the identity of the customer and the initiation of a transaction.

We analyze the two most relevant use cases, which are that of logging in and that of initiating a transaction to another account. Let us first consider the logon procedure using the two-factor authentication scheme employed by Rabobank:

Session authentication in a two-factor context

First of all, the customer enters the address of the internet banking service in his web browser. The web browser then contacts the web server of the internet banking service and requests the login page, which is subsequently transfered to the web browser and then displayed to the user. Next, the user

[9] For more information see `http://www.vasco.com/products/Digipass.html`

inserts his check card into the security calculator and enters his Personal Identification Number (PIN) code. The security calculator now computes a one time password (OTP) which is displayed to the user. Afther that, the users enters the OTP in the web browser which transmits it to the bank's web server. The validity of the OTP is checked at the bank's site. In the case of a correct OTP the user is granted access to the internet banking application. Figure 4.2 graphically depicts this procedure in a message sequence chart.

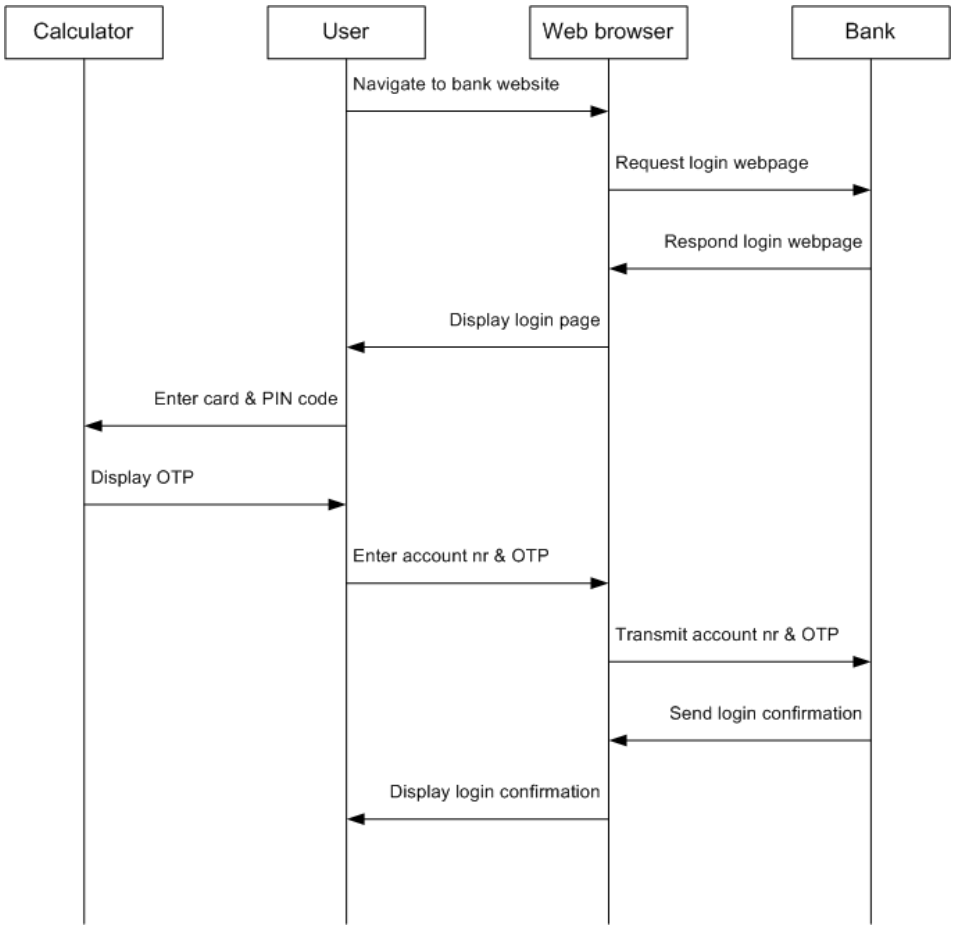

Figure 4.2: Message sequence chart of session authentication in a two-factor context

Note that if the one time passwords generated by the security calculator should only be valid within a limited timespan there must be a clock inside the calculator which is synchronized with the computer systems at the site of the bank. ABN Amro uses a slight variation of the scheme used by

Rabobank and incorporates a challenge-response action for the login action. In this scheme a challenge code is delivered to the web browser which then displays it to the customer. The customer enters his PIN code and the challenge code into the security calculator which computes a response code. The customer enters the response code in the web browser so it can be transfered to the bank's server. The response code is checked upon correctness and in case this code is valid logon is confirmed. This solves the synchronization problem, but requires the customer to perform additional operations. The login process using the ABN Amro e.dentifier is depicted in Figure 4.3.

Transaction authentication in a two-factor context

Authenticating transaction instructions is similar amongst all major banks that employ a two-factor authentication scheme. This procedure incorporates a challenge-response mechanism and goes as follows:

Once granted access to the internet banking application the customer can initiate a transaction by entering the details of the transaction (such as the amount to be transfered and the account number of the beneficiary) in the browser. Subsequently, the web browser sends this information to the server at the bank which computes a challenge code using this information. This challenge code is delivered to the web browser which then displays it to the user. The user enters his PIN code and the challenge code into the security calculator which computes a response code. The customer enters the response code in the web browser so it can be transfered to the bank's server. The response code is checked upon correctness and in case this code is valid a confirmation that the instructions for the transaction successfully arrived at the bank's computer systems is sent. This procedure is presented as a Message Sequence Chart in Figure 4.4.

For transactions that involve large amounts of money some banks introduce an additional layer of security. For example, Rabobank requires a digital signature[10] of the amount involved in the transaction when it exceeds 10.000 euro. The procedure for authenticating large transactions is depicted in Figure 4.5. Rabobank claims that for transactions that involve even higher amounts of money a second digital signature of the account number of the

[10]The term digital signature might be somewhat misleading here. This signature scheme is based on a shared secret and NOT on asymmetric cryptographic techniques. Hence, forgery of such a signature by malicious banking personnel might be possible. However, these attacks are out of the scope of our research as was specified in the threat model in Section 2.4.

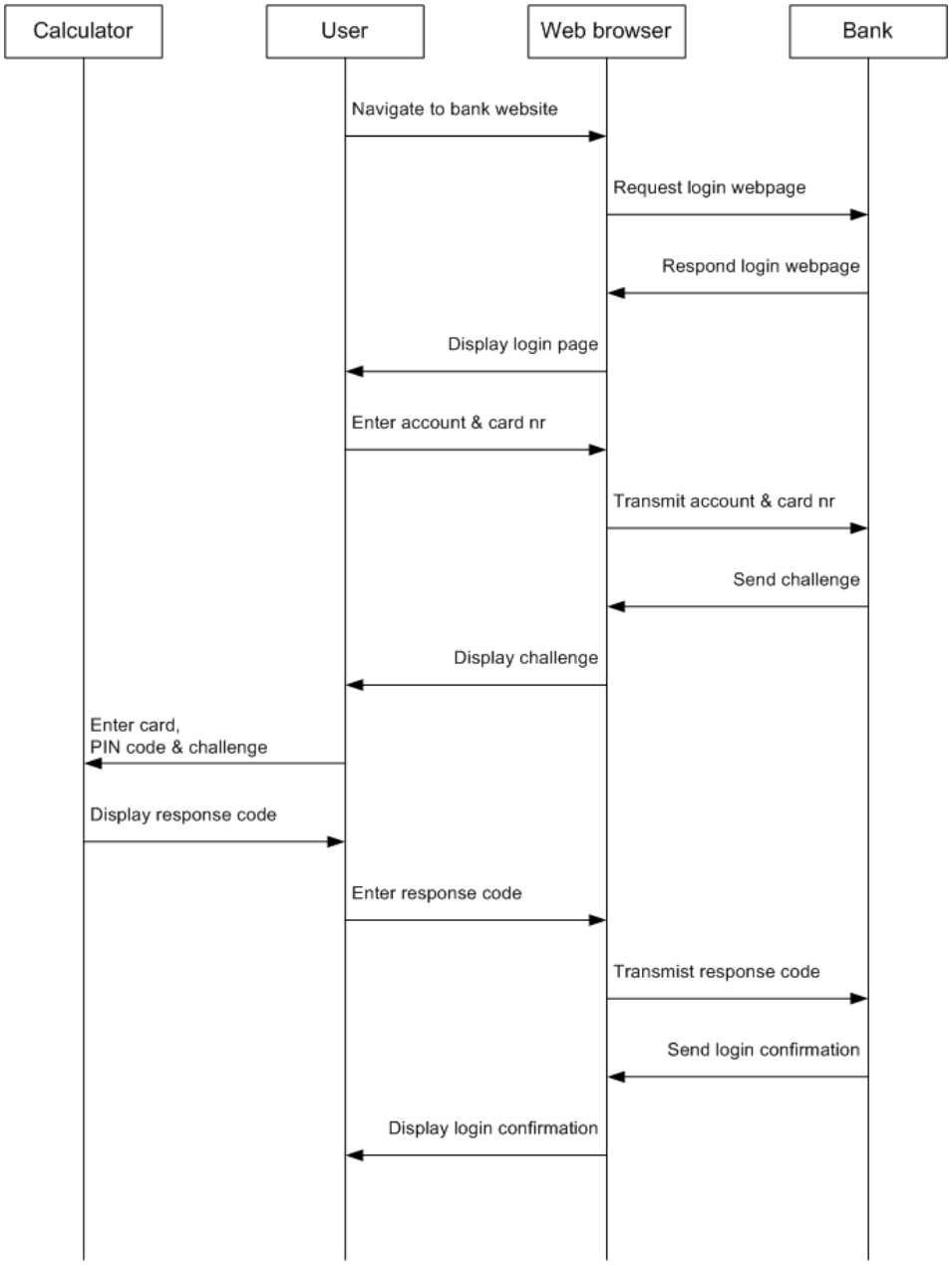

Figure 4.3: Message sequence chart of session authentication in a two-factor context at ABN Amro internet banking

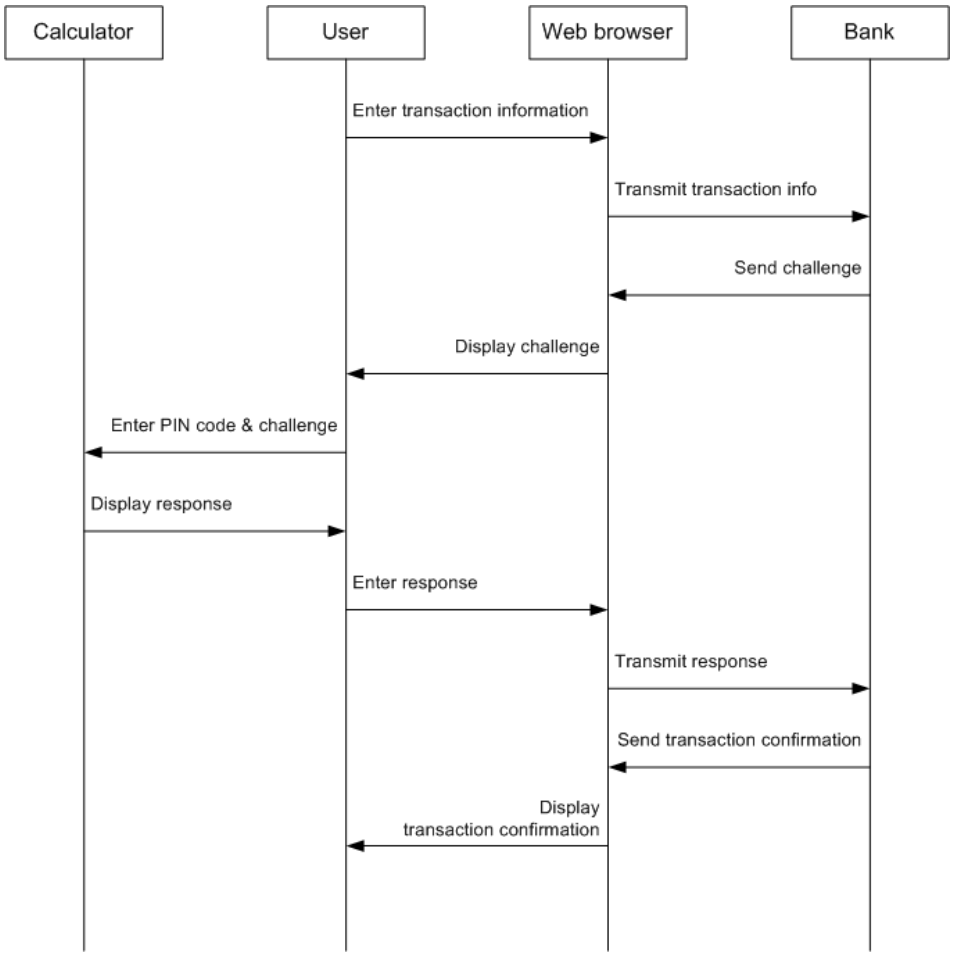

Figure 4.4: Message sequence chart of transaction authentication in a two-factor context

beneficiary is required. However, note that for transactions below 10.000 euro neither the amount nor the account number of the beneficiary is signed by the security calculator. Accordingly, for transactions below 10.000 euro the security calculator does not guarantee the integrity of this information! The calculator only creates a binding between the identity of the customer and the transaction. However, this lacks a binding between the identity of the customer and the *information* in the transaction. Hence, this is insufficient to achieve the integrity and non-repudiation requirements (see Section 2.2.1).

Two-channel customer authentication

An alternative to two-factor authentication is a two-channel authentication scheme. In such a scheme a trusted non-internet channel is used to establish authentication requirements. In the Netherlands Postbank is the only major bank that makes use of a two-channel authentication scheme[11].

Let us look at the use cases of logging in and initiating a transaction to another account in the two-channel setting of the Postbank. Strikingly, in the two-channel scheme of the postbank the trusted channel is only used to establish a verifiable binding between the identity of the customer and the initiation of a transaction. For the establish of a verifiable binding between the identity of the customer and the logon action at the beginning of an internet banking session a trivial password-based authentication scheme is used.

Session authentication in a two-channel context

As just discussed, Postbank does not use a trusted channel for logging in: The logon procedure relies on a traditional username-password based authentication scheme over a SSL encrypted internet connection. The logon procedure is presented as a Message Sequence Chart in Figure 4.6.

In the scheme of Postbank passwords have a lifetime of several months. Consequently, a phisher can replay stolen credentials.

Transaction authentication in a two-channel context

For initiating transactions via internet banking Postbank employs an authentication system that incorporates Transaction Authentication Numbers

[11] It is interesting to note that the DigiD authentication system is a non-banking initiative that supports two-channel authentication.

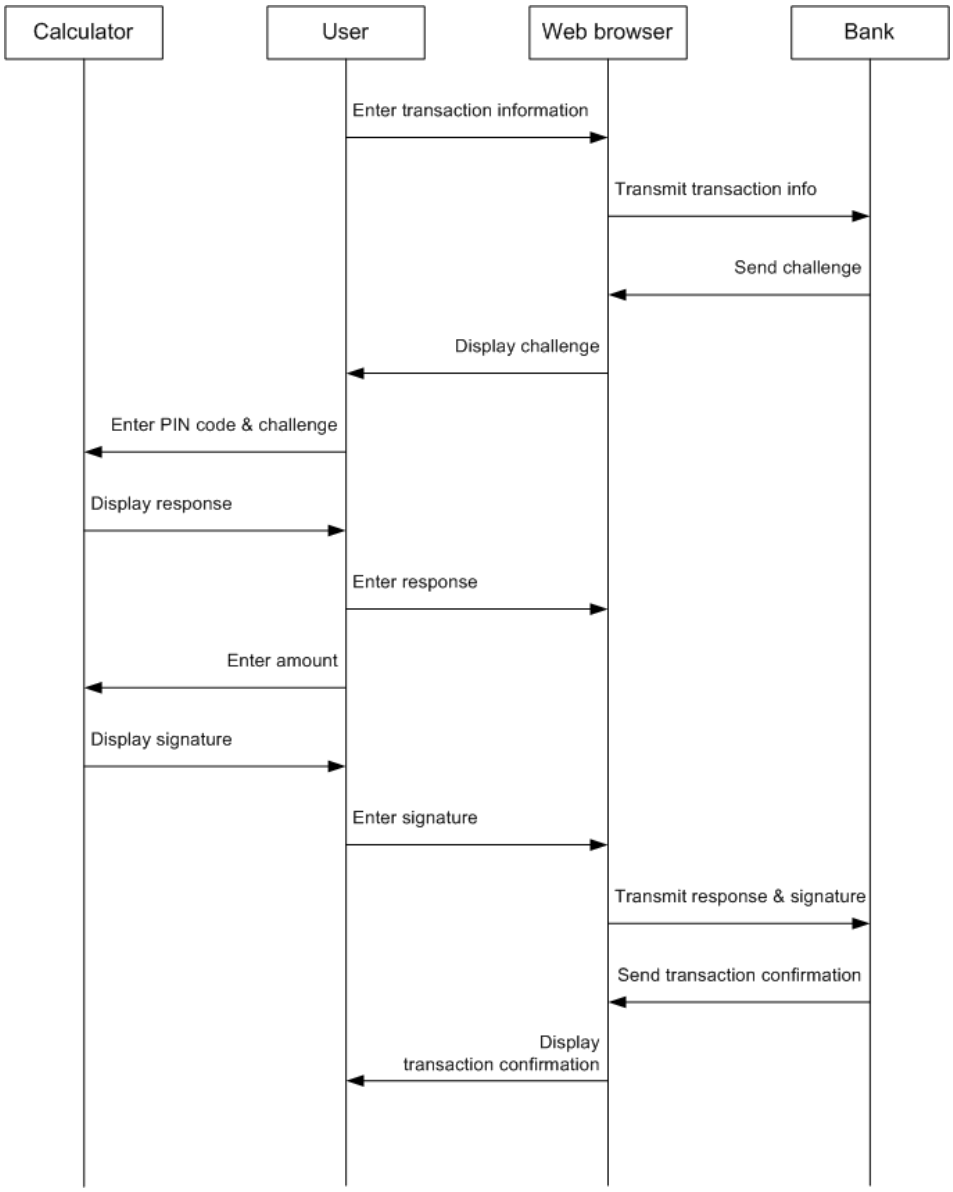

Figure 4.5: Message sequence chart of transaction authentication in a two-factor context for very large amounts at Rabobank internet banking

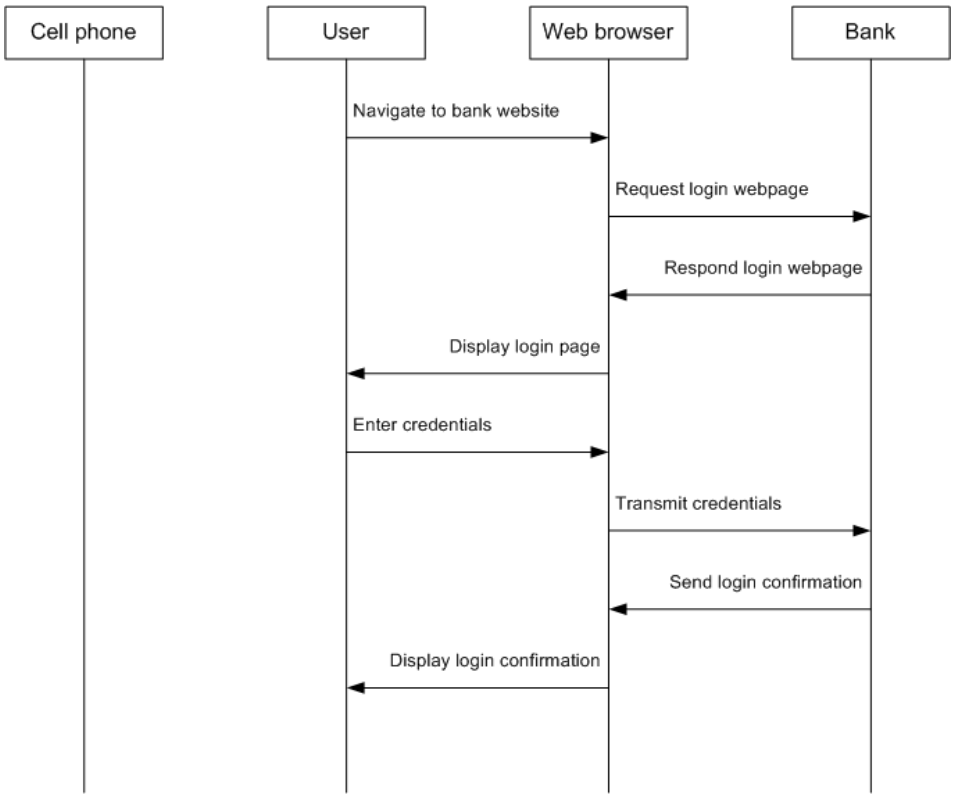

Figure 4.6: Message sequence chart of session authentication in a two-channel context

(TAN). These TAN codes serve as one-time passwords that are required to verify that the customer did actually initiate the transaction. The TAN codes are issued to the customer using a trusted channel: either a batch of TAN codes is delivered using postal service or a single TAN code is delivered using SMS text messaging when required. The Message Sequence Chart in Figure 4.7 depicts the process of initiating a transaction and receiving the TAN code via SMS text messaging.

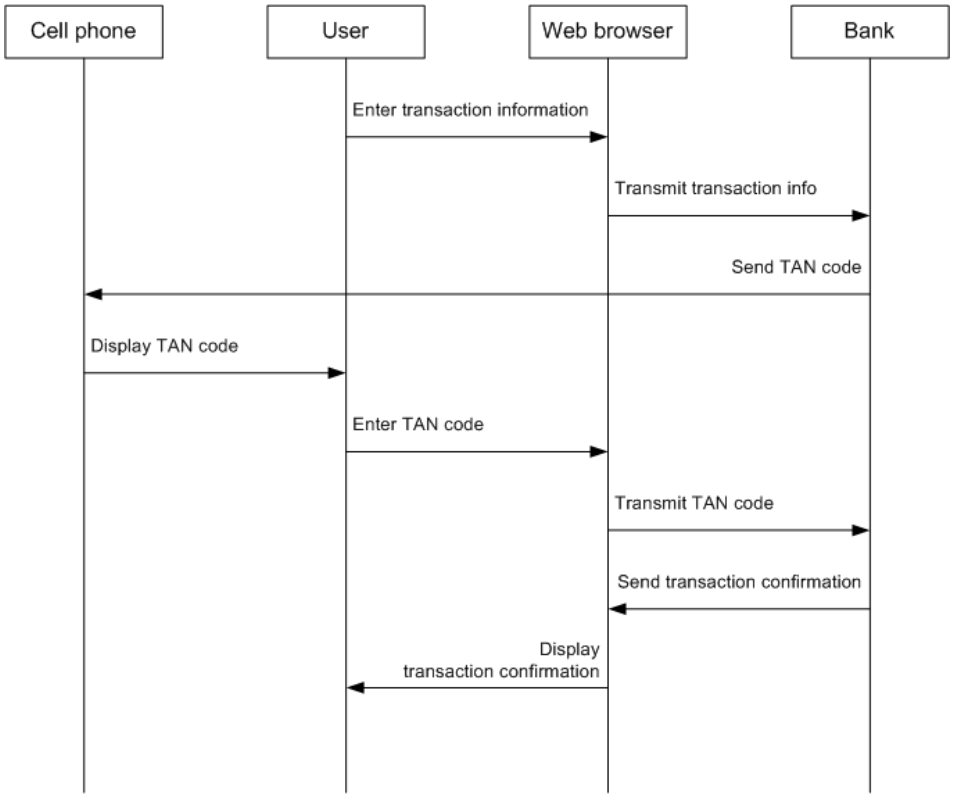

Figure 4.7: Message sequence chart of transaction authentication in a two-channel context

A central issue with TAN codes is the timespan in which they are valid. TAN codes are valid for single use but remain valid for a long time when unused (probably more than a year). In combination with the fact that TAN codes are only related to the serial number of a transaction (i.e. not related to the actual contents of a transaction) this renders them desirable objects for phishers. Once a phisher has captured valid credentials and the TAN code for the next serial number the phisher is capable of creating arbitrary transactions.

Moreover, note that TAN codes solely serve as tokens for initiating transactions. That means a TAN code is not related to the amount involved in the transaction or the account number of the beneficiary. Accordingly, when the customer enters a TAN code he does not know whether the integrity of these fields is guaranteed. In order to solve this problem banks can send a confirmation of the amount and beneficiary in a SMS text message along with the TAN code. Unfortunately, some banks lack a proper implementation of this mechanism. For example, the Postbank does not send a proper confirmation for aggregated transactions or transactions that involve an amount below 1000 euros. Hence, a trusted path from customer to bank and vice versa is assumed. That means, the integrity of the transaction depends on the correct and genuine reproduction of the transaction information of the systems (enforced by software security mechanisms) and communication channels (enforced by SSL / TLS) on the path to the bank. In Chapter 5 we will argue that this opens up attack vectors.

Let us consider to what extent the two-channel authentication based on TAN codes complies with our requirements (see Section 2.2). One important thing to note in this context is that the delivery of the TAN codes is based on a third party service provider. As far as user-friendliness is concerned this implies that the delivery is just a best effort service. This dependence can cause serious inconvenience for the customer when the third party fails to deliver the codes properly.

The dependence on third party services also raises issues on our security requirements. Since the TAN codes are delivered via third party service provider these providers should be trusted and considered safe. We made this assumption explicit in our threat model (see Section 2.4).

4.2 Back-end security solutions

In contrast to front-end security solutions, back-end security solutions are anti-phishing mechanisms that are mostly transparent to internet banking customers and which are under direct control of the bank. In the Netherlands three particular back-end anti-phishing solutions are popular: Transaction anomaly detection, log file analysis and takedowns. In the following sections these anti-phishing solutions will be extensively discussed.

4.2.1 Transaction anomaly detection

In order to detect potentially fraudulent transactions transaction anomaly detection systems are available. Transaction anomaly detection mechanisms have been in place at many financial institutions long before internet banking became popular in order to detect money-laundering practices and credit card fraud. Example transaction anomaly detection products are RSA Transaction Monitoring [18] and Actimize. According to interviews with experts all major Dutch banks operate transaction anomaly detection systems[12].

Transaction anomaly detection systems explictly counter Step 7 (Achieve pay-out) of our phishing framework. These products combine user profiling with business rules to detect suspicious account activity [31]. Suspicious transactions are alerted to the bank's professionals so appropriate reactive measures can be taken.

Transaction anomaly systems have existed for decades. As a result, these systems have evolved from a relevant history of traditional offline banking fraud and target a wide range of fraudulent transactions, including money laundering and credit card fraud. These transaction anomaly systems have been adapted to suit a new era of digital crime. As discussed in Section 3.1 the catch of a phishing attack is largely independent of the lure and the hook and leans on traditional money laundering techniques. As a result the catching phase of phishing can be targeted using these systems.

It is not surprising that transaction anomaly systems are extremely popular among Dutch banks in the fight against phishing. Banks have complete control over these in-house systems and they are completely transparent to most users. Moreover, Gartner research reports about several success stories of this form of back-end protection [62].

[12] This is not surprising. Transaction anomaly detection systems are an excellect solution to be compliant with a Dutch law on reporting of unusual transactions (Wet Melding Ongebruikelijke Transacties).

4.2.2 Log file analysis

Log analysis systems apply sophisticated and automated analysis on audit trails in order to detect phishing lures, hooks and catches. Log analysis systems are often customized software that is tailored to a specific logging situation at a bank. Interviews with experts showed that three variants of log analysis systems are most common:

- *Bounced email log analysis.* As discussed in Section 3.1 phishers often spoof the From: address in email messages to mimic the legitimate domain of the targeted bank. Consequently, when the email inbox of the victim is full or non-existent the email will bounce to the email servers of the bank. By analyzing the number of bounced messages and their content a phishing email lure can be detected.

- *HTTP referrer log analysis.* In Section 3.1 it was explained that phishers often create hooks that deep-link or anchor content of a genuine bank website. When a web browser requests such content it sends a Referrer-header that indicates the URL from which the content is linked [44]. By analyzing the Referrer-headers in the HTTP logs of the banking website phishing hooks can be identified.

- *Login log analysis.* In the impersonation phase of a phishing attack the stolen information (often credentials) is used to impersonate the victim. By analyzing the login audit trail of an internet banking application patterns can be found that raise suspicion. For example, if a large number of internet banking sessions comes from a single IP address this could be the work of a phishing casher. Additionally, by mapping IP addresses to geographical locations a bank can detect logins from countries where phishing activities are concentrated, such as Romania or Russia. An off-the-shelf product that applies this technique is the minFraud application from MaxMind.

These methods are also discussed in a report by the Identity Theft Technology Council [42]. According to this report these mechanisms can dramatically improve a bank's responsiveness to phishing attacks. Indeed, if logs are monitored in real-time, extremely quick response times could be reached. However, it is common to parse logs with a predefined interval, especially since some log analysis systems rely on cumulative statistics (for example detecting man-in-the-middle attacks by the total number of logins from a single IP address). Nevertheless, these log analysis systems allow for quick response times in a cost-effective manner. Hence it is unfortunate that interviews with experts show that not all major Dutch banks operate the three log analysis systems listed before.

4.2.3 Takedowns

After detection of a phishing website hook, for example using brand protection monitoring[13] or after a customer's report, legal actions to take the hook offline can be taken. Such actions are called takedowns and require cooperation between technically oriented personnel and legal personnel. Consequently, for many financial service providers it pays off to outsource this expertise to third-party service providers. RSA FraudAction is a well-known solutions for this.

Taking down a phishing website hook is largely based on manual labor. Hence, highly qualified technical experts and legal personnel are required for takedowns. Just as with brand protection monitoring services these costs can be spread over the user base, which renders the service affordable to large banks but expensive for smaller banks.

RSA FraudAction claims to be able to take down phishing website hooks within 5 hours on average [14]. However, experimental results show that the median lifetime of a phishing website hook is 20 hours [68].

Takedowns versus prosecution

According to experts a considerable portion of all phishing attacks is carried out by only a small number of phishing gangs [68]. Some of the experts interviewed endorse this statement, which raises the question whether the principle of takedowns is advantageous: Is it not better to target the root (phishing gangs) of the phishing problem instead of its symptoms (phishing hooks)? In 2005 Arda Gerkens, a member of the Dutch House of Commons asked the minister of judiciary questions on phishing [16]. These questions mainly considered the organization of the prosecution of phishers, which requires the digital expertise offered by the *Nationale Recherche* (National Criminal Investigation Department). However, this department only investigates heavy offenses and organized crime, which results in a chicken and egg dilemma: Proving that a phish is set up by organized crime already requires digital expertise.

[13] Third-party solutions are available for brand protection purposes [51]. These services run dedicated computer networks that spider the web and use fingerprinting and heuristic techniques to identify potential fraudulent websites and copyright misuse. Accordingly, these services can identify phishing website hooks. Prominent examples of brand monitoring service providers are MarkMonitor and VeriSign. Interviews with experts demonstrated that a considerable share of Dutch banks employed these services.

[14] One can download the RSA FraudAction data sheet at https://www.rsa.com/go/wpt/wpindex.asp?WPID=8659

Recently, there have been some arrests and prosecutions of phishers[15]. However, the criminals that were caught were just the money mules of a phishing gang while the actual criminals that organized the phishing attack (the disseminators, collectors and catchers) escaped unharmed. The police has decades of experience with prosecuting money mules as a result of traditional money laundering fraud.

On the contrary, prosecution of the criminals behind the scenes would probably require thorough investigation, considerable knowledge of digital crime and international cooperation. Fortunately, lately there has been some research from non-juridical researchers into the operating procedures and organizational aspects of phishing attacks [46][29]. Meanwhile, as prosecution procedures and cooperation have not reached the required level of maturity yet, takedowns are an effective manner to directly counter phishing attacks.

[15]This item is discussed in Dutch at `http://www.security.nl/article/17677/1/Recherche_arresteert_katvangers_virusaanval_ABN_AMRO.html`.

4.3 Evaluation of the current defensive strategy

Let us analyze the security of the defence in depth applied by Dutch banks. For this purpose we assume a scenario in which an array of common defensive mechanisms (which are discussed in the previous sections) is in place:

- *Software security.* The customer has anti-virus and firewall software installed.
- *Server authentication and link encryption.* SSL / TLS is used to authenticate the bank's website and to establish a secure connection.
- *Customer authentication.* Either a customer authentication scheme using security calculators or a scheme that incorporates TAN codes is in place.
- *Transaction anomaly detection.* The bank operates a transaction anomaly detection system.
- *Log file analysis.* The bank uses automated systems to analyze the logfiles of bounced email, HTTP referrers and access logs.
- *Takedowns.* The banks operates or employs takedown services in order to seize computer systems that are involved in phishing attacks.

4.3.1 Completeness of anti-phishing controls

In order to analyze the completeness of this set of counter measures we fit these defensive mechanisms in the framework derived in Chapter 3. We do this by depicting (see Figure 3.5) which exact steps of a phishing attack are countered by the previously described array of defenses.

One of the conclusions that one can draw from this assemblage is that the back-end systems operated by the bank cover the wide spectrum of the lure, hook and catch (described in Section 3.1) of a phishing attack. Log file analysis systems and takedowns counter phishing lures and hooks, whereas transaction anomaly detection systems cover the catch of a phishing attack.

However, although the range of the back-end systems employed by a bank is wide, these systems are quite limited in its defensive strategy. Security controls can be categorized by timing into preventive, detective and reactive (corrective) controls [11] or a combination of these. Accordingly, the controls implemented at the back-end (transaction anomaly detection, log file analysis and takedowns) are solely controls that have detective and reactive strategies.

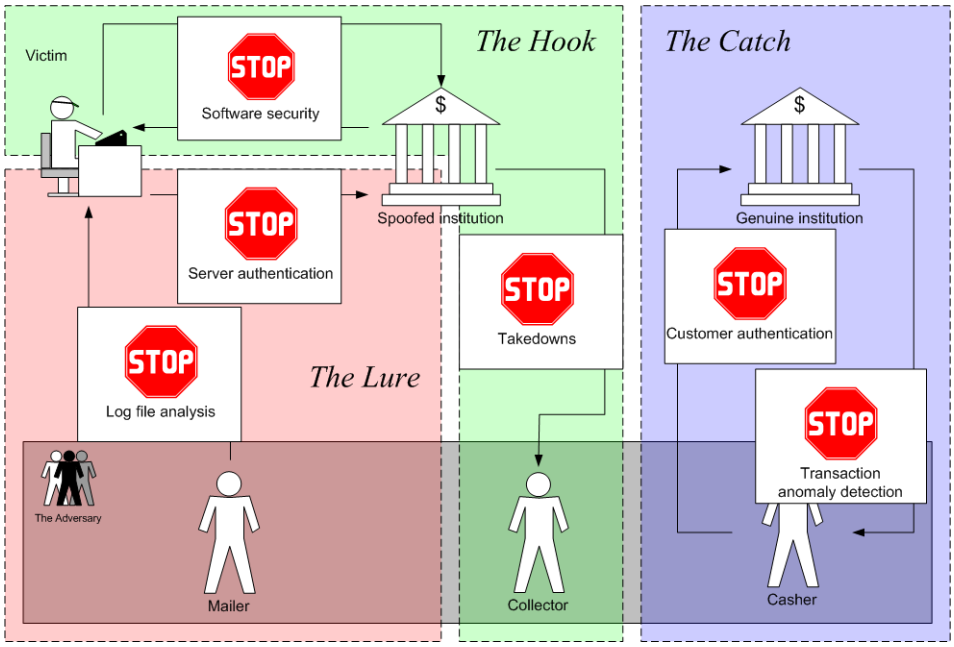

Figure 4.8: Placement of the chokepoints of the current defensive strategy in the lure, the hook and the catch of a phishing attack

The situation with the front-end software security controls is somewhat analogous. Malware scanners are a typical example of controls that have a detective and reactive strategy. Furthermore, firewalls have some preventive properties but as discussed earlier in this chapter their effect on phishing are only marginal. Hence, we can conclude that also the front-end controls do not support a preventive strategy.

Altogether, we can conclude that the preventive anti-phishing strategy of Dutch banks largely depends on either security calculators or TAN codes, depending on the authentication scheme that is implemented by the particular bank. And as the successful attack on the two-factor authentication scheme of ABN Amro (see Section 3.1) demonstrated these schemes are far from a silver bullet. Moreover, the fact that phishers launched a full-blown attack against this two-factor authentication scheme proves that there is incentive to do so and that the preventive effect of such a scheme is only limited.

Then what are the consequences of such a narrow preventive strategy? Foremost, failure of prevention and, hence, the reliance on the incremental approach of a detective and reactive strategy leads to an arms race between phishers and banks [71]. Consequently, phishers seize the opportunity to cir-

cumvent detective measures or mitigate the effects of reactive controls. For example, the introduction of fastflux technology (see Section 3.1) by phishers is a typical example of this arms race. Richard Clayton (Cambridge University, UK) expressed the need for a a sudden imposition of security controls that moves the phishers away from the online banking scenery [34]. We would like to add that such a control should have a preventive strategy in order to break the dominance of detective and reactive measures that allow the phisher to refine their techniques. In the next chapter we will evaluate the preventive effectiveness of two-factor and two-channel authentication schemes in more detail.

4.3.2 Defensibility against current attacks

Let us analyze the defensibility against current attacks using the assumed array of counter measures. The attacks described in Section 3.1 are considered: dragnet phishing, real-time man-in-the-middle and malware-based phishing attacks. We will discuss to what extent the aforementioned countermeasures offer protection against these attacks.

- *Dragnet phishing*
  Attack summary: A bulk mailing is sent to a large number of people. The contents of the email are made up of social engineering techniques that lead the victim to a website spoof in order to obtain internet banking credentials.

  Main choke points: Bounced email log analysis might be able to detect the launch of this phishing attack in an early stage. HTTP referrer log analysis might be able to detect contect used by the website spoof. Credentials generated using a security calculator are only valid within a limited timespan. The absence of a valid SSL certificate that corresponds with the bank may warn the user. Takedowns services are operated against the website spoof.

  Conclusion: Dragnet phishing is not very effective in the Netherlands. Interviews with experts demonstrated that Dutch internet banking customers are very well aware of this kind of attack. Furthermore, the limited lifetime of credentials in a two-factor authentication scheme considerably decreases their value on the black market. Hence, this form of phishing is only practical against a two-channel authentication scheme that involves TAN codes (which are valid for a much longer time). Accordingly, we think that it is not surprising that Postbank is the only Dutch bank that is frequently probed by dragnet phishing attacks. Postbank, however, claims that this statement is incorrect and

that they are a prime victim because of their large customer base[16].

- *Real-time website Man-in-the-Middle phishing*
  Attack summary: Internet banking customers are directed to a website that mimics the look and feel of a genuine internet banking service. This spoof communicates in real time with the genuine internet banking website in order to alter transactions on the fly.

  Main choke points: Login log analysis has a high chance of detecting multiple logins from a single IP address. The absence of a valid SSL certificate that corresponds with the bank may warn the user.

  Conclusion: As discussed earlier in this chapter SSL certificates are ignored by a large share of the customer base. Fortunately, a bank can detect multiple logins from a single IP address (or block). Accordingly, proper analysis of login audit trails can timely detect this class of phishing attacks.

- *Malware-based phishing*
  Attack summary: Malware is installed on the PC of an internet banking customer. The malware tries to snoop credentials and to alter transactions on the fly.
  Choke points: Malware scanners might be able to detect the malware. Firewall software might prevent the malware from transferring credentials to the attacker. Credentials generated using a security calculator are only valid within a limited timespan.

  Conclusion: Earlier in this chapter the questionable performance of malware scanners and personal firewalls in a phishing context was discussed. Accordingly, phisher know about effective ways to install malware in stealth. This leaves the malware with the challenge to evade two-factor or two-channel authentication schemes. How this can be achieved is discussed in Chapter 5.

4.3.3 Anti-phishing responsibility and liability

As discussed in the previous sections current defenses are not foolproof and have some serious shortcomings that might enable successful future phishing attack. This raises the question who is responsible in case such a successful phishing attack happens. Central issue in this debate is the importance of software security (malware scanners and personal firewalls) and the verification of SSL certificates by the customer in the line of defense against

[16]This statement is expressed at `http://www.security.nl/article/17292/1/Bankklanten_met_onbeveiligde_PC_niet_aansprakelijk.html`

phishing. We will take a look at this matter from an objective viewpoint and we will discuss our personal opinion on this subject.

ASB Bank, a major bank from New Zealand, emphasises the responsibility of the internet banking customer and even makes a liability issue of it. Since July 2007 the terms and conditions of this bank state that internet banking customers who do not have proper software security controls are not admissible for compensations in case of a phishing attack. This has serious implications that go beyond the installation of a virus scanner: ASB Bank enforces their customers to migrate to Windows Vista since full support on Windows XP will soon be dropped by Microsoft[17]. Clearly, this enforcement has serious usability and cost implications for internet banking customers.

In the Netherlands conditions are less strict. Banking professionals admitted in interviews that, altough not officially expressed, in practice customers are always compensated. Nevertheless, also Dutch banks address the importance of software security for safe internet banking use. As an outcome of this view the 3xKloppen campaign was launched by the Nederlandse Vereniging van Banken (NVB). This campaign tries to educate users on the importance of three aspects:

- Internet banking customers should maintain up to date virus scanners and personal firewalls.
- Internet banking customers should verify the genuineness of the URL of the internet banking website and the SSL certificate.
- Internet banking customers should comply with the bank's instructions for initiating transactions (e.g. use security calculators correctly and check the serial number of TAN codes).

Strikingly, this approach seems to be at variance with some of the ideas expressed earlier by the Anti-Phishing Working Group (APWG). In a document on anti-phishing controls the APWG stated that for preventive solutions to be effective end-user training should be limited as much as possible [1]. We confirm this statement, especially since the internet banking customer base is a mishmash of a variety of users. This includes elderly and technically less oriented people for whom correct interpretation of virus scanner results and verification of SSL certificates will be a hard task.

In our opinion the importance of front-end software security solutions is stressed unnecessarily in current anti-phishing strategies because of the failure of current two-factor and two-channel authentication schemes. In the

[17]This issue is discussed in Dutch at `http://www.security.nl/article/18098/1/Bank_verplicht_Vista_voor_internetbankieren.html`.

next chapter we will elaborate on this subject. We will demonstrate the weaknesses of current two-factor and two-channel authentication schemes which needlessly pressurize front-end software security controls. Moreover, we will present improvements on these authentication schemes in order to relieve software security controls in an anti-phishing strategy (see Chapter 6).

4.4 Key issues of this chapter

- Installation, maintenance and correct interpretation of front-end security controls may require computer skills that go beyond that of technically less oriented computer users.
- The anti-malware industry structurally lags behind the attackers.
- Users have a tendency to ignore the security indicators of SSL / TLS.
- Two popular variants of customer authentication mechanisms are employed in the Netherlands: two-factor and two-channel customer authentication schemes.
- Current implementations of both two-factor and two-channel customer authentication schemes lack a proper mechanism to establish the integrity and non-repudiation requirements of the information involved in internet banking transactions. This puts an emphasis on the importance of other security controls such as software security mechanisms. It is questionable whether these security controls sufficiently take care of this deficiency.
- The importance of software security mechanisms introduces liability issues.
- Transaction anomaly detection, log file analysis and takedowns are the most prevalent forms of back-end security controls.

Chapter 5

Future attack vectors

In this chapter we will combine the lessons learned from the previous chapters to perform an analysis of future phishing attack vectors. We extract key issues from our threat model (Section 2.4) and the analysis of current anti-phishing controls (Chapter 4) We will combine these results with the phishing attack framework devised in Chapter 3 in order to explore which threats and steps in our framework leave headroom for future attacks. This allows us to identify attack vectors that currently remain unexploited or are not yet explored to the fullest. Later in this chapter we will explore four examples of such advanced attacks.

5.1 Attack vector analysis

As discussed in Chapter 3 phishing gangs have become increasingly well-organized. We showed that this organization has led to a separation of concerns where the lure, hook and catch of a phishing attack are often performed by distinct entities. We think that because of the recent introduction of digital illegal marketplaces [46] this trend of specialization will continue. Eventually, this will lead to an even more distinct supply chain and a sharper distinction between the different phases of a phishing attack. Because of the strong cohesion between steps 1 to 2 (lure), 3 to 5 (hook) and 6 to 7 (catch) we believe this is a suitable subdivision into units to exame the future of phishing. We will now make a judgement of the maturity of the technology being used in the different phases of a phishing attack:

- *The lure*
  We can conclude that in the lure of a phishing attack phishers mainly adopt technologies that evolve from spamming (e.g. mass mailing), social engineering and computer viruses (e.g. drive-by installations, automated exploitation). These technologies stem from the early days of the internet and have evolved ever since. As far as reuse of existing attack technologies is concerned one can observe that the payload delivery (step 1) can be reused straightforwardly from other digital crimes such as spamming and the spread of adware. Altogether, we conclude that the technologies adopted in the lure part of a phishing attack have evolved over a considerable time and that this part of a phishing attack allows for some technology reuse of other digital crimes.

- *The hook*
  The technology used in the hook of a phishing attack is based on more recent illicit developments: Keylogging and web spoofing techniques became popular in the second half of the nineties when the free web-based e-mail service Hotmail gained massive popularity and became an attractive object for stealing credentials. Furthermore, these technologies required adaption to fit attacks on internet banking services. All in all, we conclude that the malicious technologies used in the hook of a phishing attack are tailored and fairly recent.

- *The catch*
  Compared to the technologies used in the lure and hook of a phishing attack the technologies adopted in a catch are ancient and have probably almost fully evolved. Anonymizing technologies such as anonymous proxies and Tor are used in all variants of digital crime and have proven efficient. Furthermore, the money laundering schemes employed in the last step of a phishing attack (achieve pay-out) are

Phishing phase	**Evolved from**	**Reusability**	**Maturity**
Lure	Spam, social engineering, computer viruses	Partly	Moderate
Hook	Keylogging, web spoofing	Barely	Low
Catch	Credit card fraud	Entirely	High

Table 5.1: Maturity level of phishing attack parts

the result of decaded of experience with traditional credit card fraud and other financial scams.

Our findings are summarized in Table 5.1.
Accordingly, we expect little technological progress in the catch of a phishing attack. But on the contrary, in our opinion the lure and especially the hook of a phishing attack allow for technological advancements. We expect that these advancements will be bipartite:

- *Deepening*
  We expect that future phishing attacks will deepen currently used technologies and attack methodologies. Consequently, we expect current attacks to evolve in more sophisticated attacks.

- *Broadening*
  We expect that future phishing attacks will broaden their domain. Consequently, we expect future phishing attacks to employ new technologies, targets and procedures.

In the coming sections we will use example attacks that demonstrate these development directions for both the lure and the hook of a phishing attack. First, we will glance at two state-of-the art attack vectors that happen in the lure of a phishing attack, namely spear phishing (deepening) and vishing (broadening). Subsequently, we look at an advanced phishing hook attack vector called man-in-the-browser (deepening) in more detail. Finally, we look at man-in-the-mailcient (broadening) phishing hooks.

5.2 The lure: state of the art attack vectors

In this section we will explore two state-of-the art attack vectors that are released in the lure of a phishing attack. First, we will look at a deepening attack called spear phishing. Spear phishing is an attack that tries to improve dragnet phishing technology by carefully selecting its targets. Second, we discuss vishing, a broadening attack which exploits new technology. Vishing could roughly be characterized as the equivalent of dragnet phishing over Voice over IP technology.

5.2.1 Deepening: Spear phishing

Traditional forms of phishing, such as dragnet phishing, base the propagation of their payload on mass mailing techniques derived from spamming. These techniques are intrusive and loud. Banks exploit this aspect in order to detect phishing attacks. The bounced email log analysis detection is a perfect example of a counter measure that does so. Moreover, the propagation techniques used in such attacks frequently reach many victims who are not clients at the banks for which the attack has been set up. Spear phishing attacks are phishing attacks that get around these problems by employing tailored lures in order to reach opportune victims.

The term spear phishing is often ambiguously used in both academic and industrial literature. Frequently, this term is used to denote complete phishing attacks that adopt a phishing hook based on social engineering. However, we stress that we use the term spear phishing to denote a specific type of phishing lure. By this definition it is very well possible to combine a spear phishing lure with various types of phishing hooks. We define spear phishing as follows:

A spear phishing lure is a lure in which sophisticated filtering mechanisms are employed to select and approach victims of a phishing attack.

In the rest of this section we will discuss the specific technologies and techniques being incorporated in spear phishing lures. We will also investigate the practical feasibility of such lures.

Operational technology

In order to find appropriate victims phishers can exploit the tendency to publish increasingly more private information on the world wide web. The massive popularity of social networking sites such as MySpace, Facebook, Hyves and LinkedIn confirms this issue. Unfortunately, the other side of this tendency is that the published information can easily be gathered and

exploited by attackers.

For example, the social networking site Hyves that focuses on a Dutch user base and which has gained tremendous popularity allows their users to advertise the brands they adore. Accordingly, on the URL `http://www.hyves.nl/brand/2544321/Rabobank/` a list can be found of people that express their relation with Rabobank. This is effectively a public directory that advertises almost 150.000[1] (potential) customers of Rabobank. Hyves even supports functionality to refine this list by properties such as age, gender, place of residence and marital status. Combined with the fact that internet banking use has a high density in the Netherlands [5] this list by all appearances contains ten of thousands of Rabobank internet banking customers. A phisher could easily exploit this Hyves feature to find elderly victims who probably use Rabobank internet banking services[2].

The danger of the availability of this information on the world wide web is amplified by the marketing strategies of some banks. An example of this issue is the cooperation between Hyves and Mijn Postbank: Hyves advertisements were integrated into the internet banking environment of Mijn Postbank[3]. As this fades away the borders between Hyves and Mijn Postbank it clearly opens up attack vectors for phishers.

Typically, it requires webspidering and form-filling tools to carry out a spear phishing lure. The webspider tool crawls the world wide web or particular websites for valuable information. Once information that indicates a potential victim is found the form-filling tool is used to contact the victims so the payload can be delivered.

Practical feasibility

Let us discuss the various practical aspects of a spear phishing lure.
Carrying out a spear phishing lure requires considerably more time than the propagation phase of a dragnet phishing attack. The latter can be carried out using off-the-shelf mass mailing tools. For spear phishing attacks tailored webspidering tools need to be employed. We once wrote a webspidering tool for an online advertising company that supported very similiar functionality. Development of this tool required about 40 hours of work and required only basic knowledge of HTTP and parsing. Form-filling tools are available off the shelf. Depending on the size of the target websites to spi-

[1] D.d. March 25, 2008.

[2] Of course, this example is not limited to Rabobank. Hyves also publishes lists of other banking brands.

[3] See `http://www.hyped.nl/details/20071210_unieke_samenwerking_hyves_en_postbank/`.

der the crawling process may take from several weeks up to several months. Altogether, the time span from development of the required tools to the successful propagation of the payload is in the order of several weeks.

As discussed the development of the required tools takes a reasonably experienced programmer about 40 hours of work. Therefore, development costs of 2000 US $ seem reasonable. Form filling tools are available for free on the world web. What is left is a system to run the web crawler on. For that purpose an account on a hacked system can be used (mass mailing tools for dragnet phishing run on similiar systems). Access to such systems can be bought for bargain prices at the black market [46].

Some preliminary research into the possible successes of spear phishing has been performed [54]. This research showed that a well-setup phishing attack with a spear phishing lure exploiting social networks is able to fool more than 70% of the victims while a classical dragnet approach had a success rate of only 15%. Accordingly, altough the investments required for a spear phishing lure are considerably higher than in the case of a dragnet lure the return on investment seems much larger. Especially in the Netherlands where customers are acquinted with the classical dragnet approach[4] it may pay off for phishers to employ spear phishing lures.

5.2.2 Broadening: Vishing

The term vishing is a pun in which the letter 'V' stands for Voice over IP (often abbreviated VoIP). With VoIP telephony over IP based networks is meant. Logically, vishing usually refers to the use of Voice over IP technology in phishing attacks. In order to avoind confusion we define vishing as:

Vishing attacks are phishing attacks in which Voice over IP technology is used to approach the victim.

With many VoIP services it is possible to make calls to traditional landline telephony services. Vishing lures use this feature to exploit the public trust in traditional landline telephony services while maintaining the advantages of VoIP: Voice over IP calls generally are of lower cost than traditional telephony services and they can be completely automated.

An obvious attack scenario is to make calls to a mass of victims (analogous to traditional mass mailing techniques used in dragnet phishing) using a interactive voice response (IVR) system. Using spoken social engineering

[4] Interviews with experts showed that Dutch customers are very well capable of identifying dragnet phishing lures.

texts this system can ask for credentials, TAN codes, PIN codes or other sensitive information which the system can record.

Operational technology

Research into the technology required for VoIP spamming attacks has been performed [28]. In this study a platform to carry out VoIP spamming attacks has been developed. Similiar technology can be used for automating vishing attacks. The platform consists of the open source PBX telephony engine Asterisk[5] packaged with the Trixbox[6] Linux distribution. On top of this platform the researchers built a dedicated call scheduling application called SpamScheduler that is able to make fully automated calls to a list of victims.

Such a platform needs to connect to a VoIP service network of which there are plenty available, such as Voipbuster[7]. Any internet connection will suffice in order to connect to the VoIP network. Connecting using an open WiFi hotspot or a hacked computer system provides a large degree of anonimity for the attacker.

One major implication of the technology used in vishing attacks is the marginal precense of evidence. With traditional phishing lures digital evidence of the payload (such as emails or malware code) remains on the personal computer of the victim. This evidence can be easily forwarded to experts in order to support prosecution. However, with VoIP calls this is not the case by default (unless dedicated logging mechanisms are installed) which is a significant advantage from the attacker's point of view.

Practical feasibility

Let us discuss the various practical aspects of a vishing lure.

Trixbox and PBX functionality of Asterisk are off-the-shelf software packages that require no more than a day to install and configure for an experienced Linux user. The scheduling software and the IVR system built-in to Asterisk require considerably more development and configuration work in the order of a few weeks. However, such systems are perfectly reusable and might become available in the form of toolkits, just like the Rock phish toolkit that has become available to support traditional phishing attacks. The time required to deliver the payload to all of the victims in a vishing lure depends on the number of channels that are available for making simultaneous calls. A setup that allows for three calls to be made simultaneously

[5] See `http://www.asterisk.org`.
[6] See `http://www.trixbox.org`.
[7] See `http://www.voipbuster.com`.

can reach 10.000 users in one week's time under the assumption that users are only called in daytime.

Asterisk and Trixbox are available for free since these are open source software products. Consequently, the main cost factor of setting up a vishing lure are the costs of configuring and developing the system. This requires a few weeks time and considerable expertise in VoIP systems. Hence, an educated guess of development and configuration costs of 10.000 US $ seems reasonable. Compared to these costs the costs of making calls is a bargain. Prices of making calls to Dutch landlines vary around one cent per minute. Consequently, making 10.000 calls of 2 minutes each costs only 200 US $!

It is difficult to estimate the profitability of a vishing lure. The costs of contacting people by a call or by email are almost equivalent. However, it are the setup costs of a vishing lure that impose a major disadvantage on the attacker. But since these setup calls can be spread over multiple vishing lures a vishing lure might be advantageous compared to traditional phishing lures. Especially because vishing lures leave less evidence than traditional lures which exposes the attacker to a smaller risk.

Exploiting other communication channels

Clearly, Voice over IP technology is just one example of a communication channel that can be exploited to target phishing victims. It may be wise to take a look at the development of spamming which over the years has exploited a number of communication channels like SMS text messaging, filesharing systems (e.g. Limewire), community websites (e.g. MySpace's private messages) and instant messaging (e.g. MSN Messenger). In principle, all these channels are suitable to approach phishing victims. It would be an interesting subject for future research to examine the exploitability of these communication channels in a phishing context.

5.3 The hook: state of the art attack vectors

In this section we will explore two state-of-the art attack vectors that are released in the hook of a phishing attack. First, we will look at a deepening attack called man-in-the-browser attacks. These attacks take malware-based phishing attacks to a whole new level. Second, we discuss a new class of attacks that we named man-in-the-mailclient attacks. These attacks show that not only internet banking activity but also procedures closely related to internet banking are vulnerable for phishing attacks. However, before we treat these examples we address the theory of semantic attacks.

Semantic attacks

Semantic attacks are a class of internet attacks introduced by cryptographer and security guru Bruce Schneier [83]. Schneier considers semantic attacks to be the third wave in a trend of network attacks. We will briefly consider this trend and find out what consequences it has for phishing and in particular for phishing hooks.

The second wave of attacks is entitled syntactic attacks and targets the operation logics of computers and networks. These attacks have been most prominent in the last few decades. Exploitation of software vulnerabilities and problems with cryptographic primitives and protocols are typical examples of such attacks. When Schneier classified these attacks in the year 2000, he admitted that the internet community did not yet know how to successfully protect against these attacks. But at least the problem was identified. In the last few years slight progress has been made to defend attacks syntactic attacks. For example, virtual address space randomization is prevalent on many Linux distributions and modern Windows systems support Data Execution Prevention. Of course, these solutions are no silver bullets but at least they significantly increase the effort required to perform syntactic attacks.

The third wave of attacks is semantic attacks. These attacks target the way humans assign meaning to content. Phishing hooks are a typical example of a semantic attack. Phishers exploit the human-computer interface by the fact that a human cannot distinguish a legitimate electronic transaction from one in which a malicious party is involved.

In Section 4.1.2 we discussed that current implementations of two-factor and two-channel authentication do not protect the information involved in a transaction. In the coming sections we will discuss two attacks that exploit this issue. We show how to let a customer believe that he is involved in a

legitimate electronic transaction while actually a malicious party is involved

5.3.1 Deepening: man-in-the-browser attacks

Man-in-the-browser attacks are attacks that recently shocked internet banking experts. Rumours arouse that sophisticated malware was able to maliciously modify browser behavior and to evade two-factor authentication. Only preliminary research into this attack has been performed [49][88]. In this section we will expose the fundamental vulnerability that enables this attack. Moreover, we will demonstrate the practical feasibility of this attack by developing full-featured exploit code. In the next chapter we will propose a defensive solution that can protect against this and other attacks.

Problems with two-factor authentication

At its introduction two-factor authentication was considered to be a silver bullet against phishing attacks. However, as the real-time man-in-the-middle attacks on ABN Amro pointed out these systems are not able to fully counter phishing attacks. In this section we will address the security issues of multifactor authentication and thoroughly explore an attack vector that exploits these issues.

In order to clarify the security issues considering current multifactor authentication implementations we will recall the use case of initiating a transaction in a two-factor authentication scheme (as explained in Section 4.1.2). The following principals are involved in such a connection.

- A(lice): Customer
- B(ank): Bank webserver
- C(alculator): Security calculator
- W(eb browser): Web browser installed on customer's PC

The following protocol represents the communication that takes place:

1. $A \rightarrow W : Account, Amount$
2. $W \rightarrow B : \{Account, Amount\}_{SSLkey}$
3. $B \rightarrow W : \{Challenge(Account, Amount)\}_{SSLkey}$
4. $W \rightarrow A : \{Challenge(Account, Amount)\}_{SSLkey}$
5. $A \rightarrow C : Challenge(Account, Amount)$
6. $C \rightarrow A : \{Challenge(Account, Amount)\}_{SKc}$
7. $A \rightarrow W : \{Challenge(Account, Amount)\}_{SKc}$
8. $W \rightarrow B : \{\{Challenge(Account, Amount)\}_{SKc}\}_{SSLkey}$
9. $B \rightarrow W : \{Account, Amount\}_{SSLkey}$
10. $W \rightarrow A : Account, Amount$

1. The customer enters the details of the transaction in the web browser.
2. The web browser transmits the transaction details to the bank's webserver via the SSL tunnel.
3. A challenge of the transaction details is sent via the SSL tunnel from the bank's webserver to the web browser.
4. The web browser presents the hash to the customer.
5. The customer enters the hash in his security calculator.
6. A response code in the form of a digitally signed version of the hash is presented to the customer.
7. The customer enters the response code in the web browser.
8. The web browser forwards the response code via the SSL tunnel to the bank's web server.
9. The web server sends a confirmation of the transaction to the web browser via the SSL tunnel.
10. The confirmation of the transaction is displayed to the customer.

The goal of this protocol is to communicate the details of the transaction securely (i.e. in conformance with the requirements derived in Section 2.2). This communication happens inside an internet banking session that has already been established. Accordingly, amongst others the integrity requirement should be met according to the requirements derived in Section 2.2.1:

- Integrity: The transaction information and its confirmation arrive at the receiving party as intended by the sending party.

Clearly, a run of the aforementioned protocol should establish this requirement. In order to do so the two-factor scheme incorporates a challenge-response mechanism. A notable property of current implementations is that a customer only enters the challenge into the security calculator. As the relation between this challenge and the transaction information (account number and amount) is not visible to the customer it is assumed that the integrity of the transaction information is guaranteed in the communication between customer and bank. Hence, a trusted path from customer to bank and vice versa is assumed. That means, the integrity of the transaction depends on the correct and genuine reproduction of the transaction information of the systems and communication channels on the path to the bank.

The attack

Unfortunately, this assumption conflicts with the assumptions we made in our threat model. Namely, tampering with the personal computer of the customer was identified as a high-risk threat. Accordingly, it is reasonable to model a new principal in the form of a piece of malicious code that can take over the software running on this personal computer, including web browser behaviour:

- E(vil browser): Web browser controlled by adversary

As the web browser is an essential chain in the trusted path from the customer to the bank this malicious code allows to the attack to infiltrate in this path. This opens up an attack vector that enables an attacker to break the integrity requirement. Namely, the attacker can perform an attack in which the browser under malicious control spoofs a genuine transaction to the customer. Meanwhile, in communication between the browser under malicious control and the bank a transaction in favor of the attacker is communicated. We can model this attack by adapting the previously described protocol. In this protocol the genuine browser principal (W) has been replaced by the principal that models a browser under control of the attacker (E). This evil web browser exploits the fact that there is no relation between the challenge-repsonse mechanism and the transaction information:

1. $A \rightarrow E : Account, Amount$
2. $E \rightarrow B : \{EvilAccount, EvilAmount\}_{SSLkey}$
3. $B \rightarrow E : \{Challenge(EvilAccount, EvilAmount)\}_{SSLkey}$
4. $E \rightarrow A : \{Challenge(EvilAccount, EvilAmount)\}_{SSLkey}$
5. $A \rightarrow C : Challenge(EvilAccount, EvilAmount)$
6. $C \rightarrow A : \{Challenge(EvilAccount, EvilAmount)\}_{SKc}$
7. $A \rightarrow E : \{Challenge(EvilAccount, EvilAmount)\}_{SKc}$
8. $E \rightarrow B : \{\{Challenge(EvilAccount, EvilAmount)\}_{SKc}\}_{SSLkey}$
9. $B \rightarrow E : \{EvilAccount, EvilAmount\}_{SSLkey}$
10. $E \rightarrow A : Account, Amount$

1. The customer enters the details of the transaction in the web browser.

2. *The evil web browser changes the transaction details and transmits these to the bank's webserver via the SSL tunnel.*

3. *A challenge of the modified transaction details is sent via the SSL tunnel from the bank's webserver to the evil web browser.*

4. The evil web browser presents the hash to the customer.

5. The customer enters the hash in his security calculator.

6. A response code in the form of a digitally signed version of the hash is presented to the customer.

7. The customer enters the response code in the evil web browser.

8. The evil web browser forwards the response code via the SSL tunnel to the bank's web server.

9. The web server sends a confirmation of the transaction to the evil web browser via the SSL tunnel.

10. *A confirmation of the original transaction is displayed to the customer.*

The following message sequence chart in Figure 5.1 depicts a man-in-the-browser attack in a two-factor context.

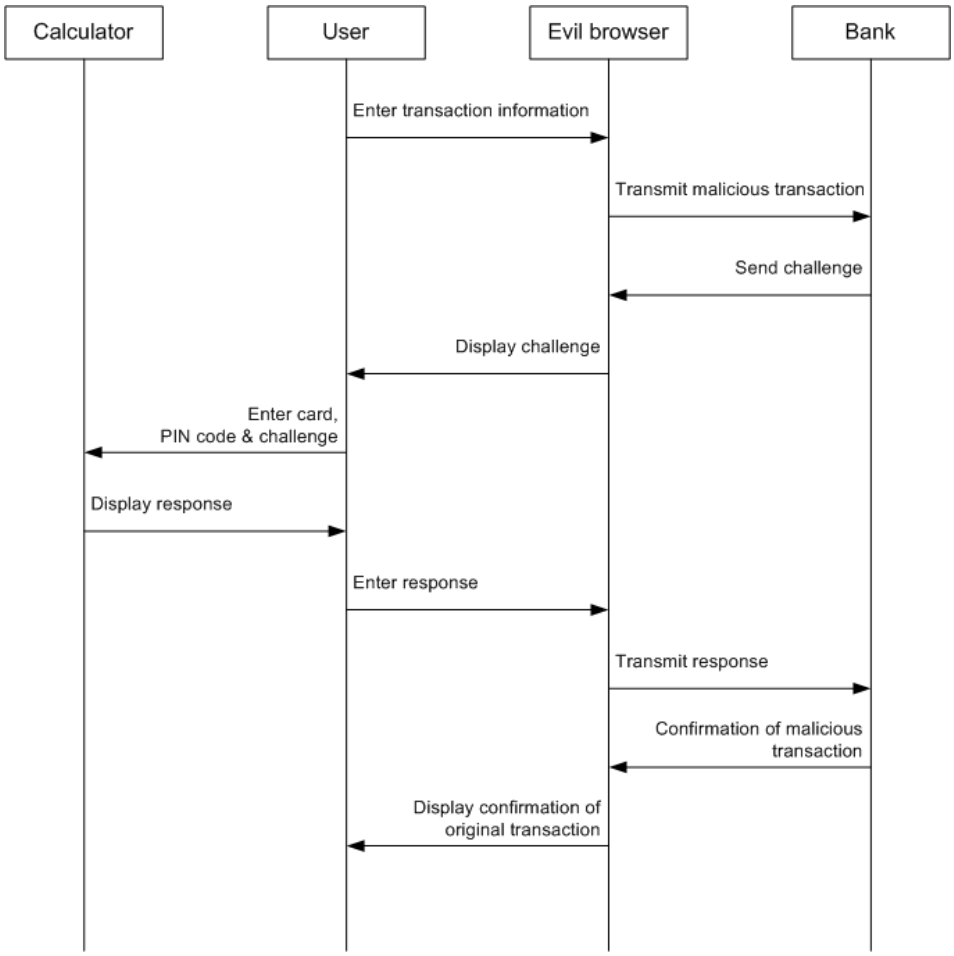

Figure 5.1: Message sequence chart of a man-in-the-browser attack in a two-factor context

Vulnerability of the two-channel scheme

Unfortunately, the two-channel scheme discussed in Section 4.1.2 is also vulnerable for the man-in-the-browser attack. Again, it is the dependence on a trusted path that causes this problem: There is no relation between a TAN code and the transaction information (account number and amount). Thus, it must be assumed that integrity of the transaction information is guaranteed in the communication between customer and bank. Hence, a trusted path from customer to bank and vice versa is assumed. That means, the integrity of the transaction depends on the correct and genuine reproduction of the transaction information of the systems and communication channels on the path to the bank. As just discussed this assumption is unrealistic since customers' end-systems are easily compromised.

The message sequence chart in Figure 5.2 depicts the concept of a man-in-the-browser attack in a two-factor context.

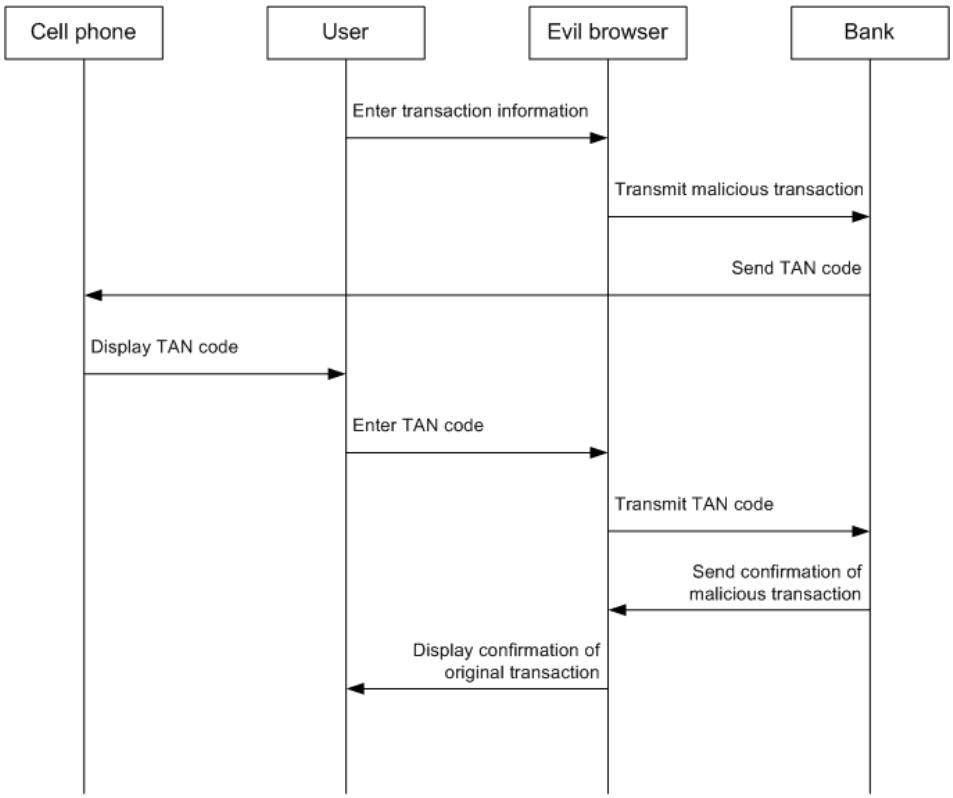

Figure 5.2: Message sequence chart of a man-in-the-browser attack in a two-channel context

Man-in-the-browser proof of concept

For demonstration purposes we developed a fully working proof of concept that can perform a man-in-the-browser (MitB) attack on two popular Dutch banks: one that employs a two-factor authentication scheme and one that employs a two-channel authentication scheme. In this section we will show some relevant technology that we used to build our attack. However, we refrain from a step-to-step tutorial in order not to support illicit activity. Demonstrations of our proof of concept are available upon request. Requests for source code will be denied.

Note that we only implemented the hook and a trivial catch of a man-in-the-browser phishing attack. For the lure there are plenty of well-known techniques to spread the MitB malware, such as drive-by-installations (see Section 3.1). The lure of our attack consists of redirecting a single transaction to our bank account.

Let us now briefly discuss the technology used to implement this attack.

Browser Helper Object

We implemented our proof of concept as a Browser Helper Object (BHO). A BHO is a Dynamic Link Library that is registered in the Windows Registry. Upon startup the Internet Explorer web browser reads this registry key and loads the corresponding module. Examples of good-natured BHO's are the Google Toolbar and Adobe Acrobat plugin. On the contrary, our proof of concept is a malicious plugin.

Once our DLL is loaded by the web browser we access its Document Object Model [22] that allows us to control the currently loaded web page (using the `IHTMLDocument2` interface) and navigation behavior (using the `IWebBrowser2` interface). We exploit this functionality by creating a hook on the `Navigate2` method. This means our hook is called every time a navigation action takes place. Our hook detects navigations that try to initiate a transaction at one of our targets' internet banking website.

Once such a navigation was detected we cancel the navigation and replace it by a navigation that actually initiates a new transaction in our favor[8]. Meanwhile, we store the original transaction information locally on the victim's harddisk. Whenever the bank's web servers send a web page that

[8] This means we can choose to initiate a transaction of any amount to any account number.

contains a hint to the modified transaction (e.g. the overview of the customer's latest transactions) we replace this text and display the information of the original genuine transaction instead. This information remains on the hard disk so the substitution can also be made in future internet banking sessions. Accordingly, the scam will only come out when the customer receives a paper print of his bank statement. In the Netherlands the major banks send such a paper bank statement monthly[9].

Man-in-the-browser attacks are extremely feasible. All of the aforementioned functionality requires less than 1000 lines of code in the Delphi programming language. Our proof concept makes uses of a publicly available BHO library which considerably cuts development time. Only moderate knowledge of Windows programming and at most three weeks time is required to implement this attack. As a consequence, we estimate that hundreds of thousands of programmers worldwide are capable of developing code that is equivalent to the code we wrote. In any case, there is current malware in circulation (e.g. the *Storm worm*) that is considerably more complicated than the code required to perform a man-in-the-browser attack. In our opinion, this underlines how plausible the threat of man-in-the-browser attacks is. Hence, we think man-in-the-browser attacks are a serious threat that require a solution on a short term. In the next chapter we propose such a solution.

5.3.2 Broadening: Man-in-the-mailclient attacks

Semantic attacks do not necessarily target the transaction initiation procedure. Phishers can broaden their domain and perform attacks targetting other procedures involved in internet banking. Man-in-the-mailclient attacks are a typical example of such attacks. We consider man-in-the-mailclient the e-mail client equivalent of man-in-the-browser.

We are not aware of any material that addresses these attacks. Nor have we seen anything like this in the wild. Probably, we are the first to document on this class of attacks.

Attacking the road towards a transaction

Before a customer initiates a transaction, an incentive to do this activity must have preceded. For example, a webshop might have sent an invoice to the customer or a friend of the customer might have sent a message that the customers has debts. We say that the party that requests money from the customer sends *payment instructions* to the customer.

[9] By default. For additional charges it is possible to receive a paper print of the bank statement more frequently.

Often, these payment instructions arrive at the customer via e-mail. For example, many smaller webshops that do not have an automated payment processing webservice simply send the customer an invoice via email. Since the customer just ordered a good via the webshop he will probably not question the authenticity of the invoice. However, how does the customer know that the contents of the invoice are genuine? This opens up an attack vector for a semantic attack that is analogous to the man-in-the-browser attack.

An attacker could develop malware that takes over the control of the e-mail client in a similiar way as the man-in-the-browser malware did with the web browser. As a matter of fact, many modern HTML-enabled e-mail clients use browser components to render e-mail. As soon such as the man-in-the-mailclient detects an email that contains payment instructions the malware changes these instructions (for example the account number) to the advantage of the attacker.

Recognizing payment instructions

This leaves the attacker with the challenge to recognize payment instructions. We will now discuss one trick that allows an attacker to do this easily.

In the Netherlands the *elfproef* is a checksum technique that is adopted by the majority of the banks. A simple calculation allows to check whether a number that consists of 9 or 10 digits is a valid bank account number. As a consequence, the man-in-the-mailclient could check every incoming e-mail message for numbers of this length and calculate the elfproef on it. When the elfproef succeeds the malware changes the representation behavior of the mailclient such that the account number is replaced by an account number that diverts the money to the attacker.

Banks' responsibility and reactions

Technically, the procedure of sending payment instructions is outside the scope of internet banking. However, because of the strong cohesion between digital payment instructions and the initiation of internet banking transactions, successful man-in-the-mailclient attacks could greatly undermine confidence in electronic payments and therefore also influence internet banking.

Recently, Dutch banks seem to emphasize the need for a convenient and secure procedure to handle digital payment instructions. Interviews with experts confirm this issue. We see a trend that the focus of internet banking

security is broadened to digital payments. The introduction of iDeal[10] is a practical example of this issue. Unfortunately, currently this trend mainly focusses on incorporating merchants and does not fix the issue for payment instructions exchanged between individuals.

What is required is a mechanism that applies the same level of security that is required for initiating internet banking transactions to digital payment instructions. That means we need a mechanism that can protect all crucial banking activity and not only the initiation of transactions. In the next chapter we propose a solution that is capable of doing so. This solutions allows a creditor to sign a *request* for a transaction. The creditor could send this request to the bank of the debtor. This bank verifies the signature which guarantees the authenticity and integrity of this request and then asks the debtor to confirm this transaction request and subsequently initiate the transaction. Roughly speaken, our solution hereby enables a secure online equivalent of the traditional transfer form.

[10]See `http://www.ideal.nl` for more information.

5.4 Key issues of this chapter

- We conclude that the evolution of phishing attacks will take place especially in the lure and hook.
- We think that the developments of future attacks will be bipartite. On the one hand we will see the employment of new technology (broadening) and on the other hand we will see more sophisticated exploitation of current technologies (deepening).
- An example of a deepening future attack in the lure is spear phishing. In spear phishing only a smaller number of interesting victims is selected which allows attackers to achieve a higher return on investment and to create more stealthy phishing attacks.
- An example of a broadening future attack in the lure is vishing. Vishing lures, in which Voice over IP technology is used to reach phishing victims, are an affordable alternative to traditional dragnet lures. Other communication channels such as instant messaging and social networking sites might also be exploited to carry out phishing lures.
- We expect the current malware-based phishing attacks to evolve into man-in-the-browser attacks (deepening), which are attacks that redirect transaction initiations using a malware-infeced web browser. We demonstrated the practical feasibility of such an attack. Man-in-the-browser attacks can be realized at low costs and with moderate skills.
- We expect future phishing hooks to target other internet banking activity than just the transaction initiation activity (broadening). We described man-in-the-mailclient attacks, a class of attacks that successfully exploits the transaction request procedure.

Chapter 6

An enhanced defensive strategy

In this chapter we propose a defensive strategy that we believe to be able to withstand all current phishing attacks and the most important future phishing attacks on internet banking services. In order to do so we will extract the key issues from our requirements on internet banking (Section 2.2), our threat model (Section 2.4) and our vision on the future of phishing attacks (Chapter 5). Using this knowledge we are able to contrive a list of requirements for our defensive strategy. Subsequently, a conceptual defensive solution in the form of a signing application is derived from these requirements. Taking the current defensive strategy (Chapter 4) into account we will discuss several considerations for the practical implementation of our proposed solution.

6.1 Conceptual defensive solution

In this section we present a conceptual solution to defend against current and future phishing attacks. Note that we first sketch an ideal and theoretical solution free from any practical constraints and independent of any implementation technology. Only in the next section we look at practical considerations to achieve the best implementation that suits practical constraints. Our solution targets the hook and the lure of a phishing attack: we render stolen information invaluable and we make the modification of information infeasible (which counters the impersonation step of the catch).

6.1.1 Relevant observations

Clearly, we want our solution to fit the ideas and issues observed earlier in this report, such as our threat model and current and future phishing attack patterns. Therefore, we will summarize the key issues based on earlier observations in this report that should be taken into account when developing our defensive solution. For each of these observations we express the consequences for our anti-phishing solution:

- *Customers' end-systems are compromised*
  From our threat model in Section 2.4 we can conclude that tampering with customers' end-systems (e.g. personal computers and mobile telephones) poses a medium to high risk threat depending on the technology involved. Moreover, in Section 4.1 and Section 4.3 we have demonstrated that this threat is not sufficiently mitigated by software security mechanisms. *Hence, our defensive solution should not rely on the security of these end-systems.*

- *Resistance against improper usage*
  In Section 4.1 we have concluded that customers regularly fail to properly interpret security indicators either because of ignorance or indolence. For example, SSL / TLS implementations do not fit customers' capabilities and behaviour which leads to the successes of real-time man-in-the-middle phishing attacks. *Consequently, we conclude that our solution should present a simple and convenient interface to the customer that can not be disregarded.*

- *What you see is what you sign*
  In Chapter 5 we elaborated on phishing attacks that are likely to arouse in the coming years. There we have demonstrated the likelihood of attacks that exploit the possibility to alter the presentation of data, such as Man-in-the-Browser attacks. This vulnerability is caused by the lack of a possiblity for the customer to confirm that

their data was received correctly. *That is why our solution should allow customers to express their consciousness and the authenticity of their internet banking activity and to confirm the correctness of the information involved in this activity.*

6.1.2 Elaboration of our defensive concept

How do we realize a defensive concept that meets these observations? We will now form ideas that suit these observations. In order to not be restrained by practical limitations we present these ideas in the form of general techniques unrelated to specific hardware or software technology.

Trusted computing base

First of all, the observation that *customers' end-systems are compromised* implies that our security critical components can not run in the compromised environment of these end-systems. We solve this by introducing a trusted computing base (TCB)[1] [9]. This TCB is an entity consisting of hardware and software that is designed in such a way that exploitation of it is extremely unlikely. This allows us to design a system in which the security of internet banking activity does not depend on the security of the customer's end-systems [61]. Note that the TCB is critical for the security of our concept. Accordingly, when the TCB is broken our system will likely be broken too. Therefore, the less complex the TCB is, the better it is for the verification of the security of the TCB. Moreover, we require that the TCB can not be operated remotely (e.g. from the internet). This requires physical access to the TCB, which we considered out of the scope for phishing attacks (see our threat model in Section 2.4).

End-to-end security

By the aforementioned observation in combination with the observation of *what you see is what you sign* we conclude that this TCB should enable end-to-end security. That means from the trust boundary of the banking systems (see Section 2.4) as close as possible to the customer (i.e. without intervention of a non-trusted entity) and vice versa. It is not the customer's end-systems that should be regarded as a safe end-point for communications as is the case in current internet banking contexts and which enables Man-in-the-Browser attacks. Hence, the TCB should be under direct physical control of the user, without intervention of a possibly compromised end-system and without dependence on the correct and genuine reproduction of data by these end-systems.

[1] Note that this is not the same as the term trusted computing as used by the Trusted Computing Group, see `https://www.trustedcomputinggroup.org`.

Digital signatures

In the observation of *what you see is what you sign* we have expressed the need for a mechanism that allows customers to express their consciousness and the authenticity of their internet banking activity and to confirm the correctness of the information involved in this activity. We can divide this issue into two aspects, namely authenticity and integrity:

- *Authenticity:* Our concept should allow a customer to create a non-forgeable binding between his identity and the internet banking activity and the corresponding information involved that can be verified by the banking systems[2].
- *Integrity:* Our concept shoud allow the banking systems to verify that information involved in internet banking activity was received correctly and as intended by the customer.

We will rely on cryptographic techniques[3] that enable digital signatures in order to achieve these goals. These techniques must be implemented on the TCB as the security of our system heavily relies on these signatures. In order to create a binding between the identity of the customer and the generation of a digital signature this action should require the entry of a PIN code by the customer. Furthermore, a timestamp (or counter) can be generated by the TCB in order to guarantee freshness of the information.

Non-neglible user interface

Our observation of *resistance against improper usage* expressed the need for a mechanism that can not be disregarded. We do so by making the digital signature action an obligatory step before crucial internet banking activity (e.g. initiating a transaction, logging in or changing one's correspondence address) can be completed. Creating a digital signature should be a conscious user action. The requirement to enter a PIN code before being able to generate a digital signature will likely support this awareness.

Summary of our defensive concept

Altogether, we summarize our defensive concept as follows:

[2] A sidenote for the understanding of the reader: Note that in Chapter 4 we observed that two-factor authentication mechanisms are able to establish a verifiable binding between internet banking activity and the identity of the customer. However, these mechanisms lacked a proper manner to establish a verifiable binding between the information involved in this activity and the identity of the customer.

[3] Altough we do not yet point to implementation technology an obvious choice for these cryptographic techniques would be digital signatures based on asymmetric crypto. These techniques establish non-repudation by enabling a third-party (e.g. the legal court) to verify any signature made [66].

We propose a signing application running in a trusted computing base only available under direct physical control of the customer that requires a customer to digitally sign every crucial internet banking activity and the relevant information involved in this activity.

The message sequence chart in Figure 6.1 depicts the operation of our concept on an abstract crucial internet banking activity (for example, imagine *activity* is a transaction).

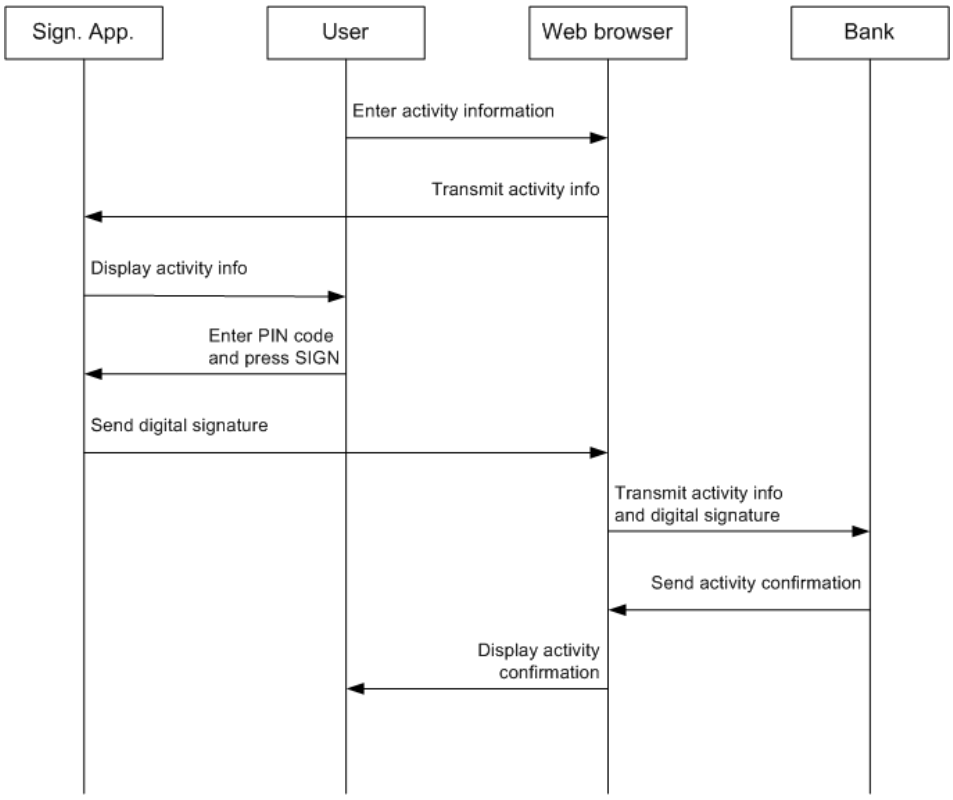

Figure 6.1: Message sequence chart of the operation of our concept on an abstract crucial internet banking activity

6.1.3 How this matches our internet banking requirements

In the previous chapters we have seen how current and future phishing attacks are able to violate our internet banking requirements (see Section 2.2) and how the current defensive strategy fails to sufficiently mitigate this. For example, real-time Man-in-the-Middle phishing attacks are able to violate the integrity and authenticity requirements that we derived in Section 2.2.1.

Let us now discuss whether our defensive solutions establishes the internet banking requirements and whether it is any better than the existing defensive strategy discussed in Chapter 4.

We recall the security requirements and briefly discuss the relevant aspects of our proposed solution (non-security requirements are dealt with when treating the implementation issues):

Security requirements

- *No-intrusion*
  No-intrusion is established since only messages that have a correct digital signature of the customer are accepted by the bank.

- *Authenticity*
  The TCB can only be operated under direct physical control. In Section 2.4 we defined that such access is out of the scope of our adversary. Moreover, only the customer knowns the PIN code to access the application. Hence, this allows the bank to verify the identity of the customer who sent a signed message. Our solution does not provide any mechanism to allow the customer to verify the authenticity of the banking systems.

- *Freshness*
  A timestamp generator inside of the TCB guarantees the freshness of every message signed using the signing application.

- *Confidentiality*
  No additional measures in order to guarantee confidentiality of messages are taken in our proposed solution.

- *Integrity*
  Our signing application requires that the generation of a digital signature happens with full consciousness of the customer. Moreover, this is done solely trusting the TCB (without dependence on the correct and genuine reproduction of data by the customer's end-systems). Hence, this mechanism allows the banking systems to verify that information is received as was intended by the customer.

- *Non-repudation*
  The TCB can only be operated under direct physical control and only the customer knowns the PIN code to access the application. Moreover, our signing application requires that the generation of a digital signature happens with full consciousness of the customer and does not rely on non-trusted systems. Hence, a customer can not plausibly deny internet banking activity that was signed for with his digital

signature. Our solution does not provide a mechanism to establish non-repudation for activity carried out by the bank.

- *Accountability*
  The digital signature mechanism allows the bank to trace crucial internet banking activity back to a unique customer.

Altogether, not all of our internet banking requirements are met. We had to make some trade-offs that affected the establishment of these requirements. As a result, confidentiality is a requirement that is not established by our solution. Furthermore, the authenticity and non-repudation requirements need some additional arguments in order to understand why they are not fully met. Let us discuss these issues:

- *Lack of confidentiality*
  We believe that a system in which all private information involved in internet banking activity (such as account balance and transaction information) is only disclosed to the customer by the TCB and vice versa is infeasible. Such a system would require that the capabilities of our proposed TCB were greatly extended. For example, a secure monitor and keyboard would be required. Not only does this open up the possibility of vulnerabilities in our TCB, it also has serious consequences for the user-friendliness and roaming requirements. Consequently, we believe that input and output of private information should remain a task to be done on the customer's end systems (e.g. personal computer or mobile phone) because of convenience reasons and to allow for a simple and small TCB. As a result of this trade-off, we turned confidentiality into a requirement that is subordinate to the other requirements derived in Section 2.2. In order to achieve a modest level of confidentiality we stick with current controls to protect against disclosure of private information involved in internet banking activity, such as SSL / TLS and software security mechanisms. Unfortunately, these controls are not foolproof as demonstrated in Chapter 4.

- *Lack of mutual authentication*
  A characteristic of internet banking is that all crucial internet banking activity is initiated by the customer. Moreover, it is the customer who sends crucial information to the bank's computer systems and not the other way around. Accordingly, if the integrity and authenticity of the information that is sent from the customer to the bank is sufficiently protected this information is invaluable for an attacker. Consequently, we do not see a need to provide a mechanism for authentication of the bank to the customer.

- *Lack of non-repudation for the bank's activity*
  By tradition, banks are trusted entities. This is exactly why customers

entrust banks to administer their money. Moreover, in our threat model we expressed our trust in the security of the bank's computer systems. As a consequence, we do not require non-repudation for the bank's activity.

6.2 Implementation considerations

In this section we will discuss alternatives for implementing our proposed signing applicaton. We believe these alternatives to be most feasible.

First, we present a short-term solution based on the current defensive strategy, which only implements our concept by approximation. Altough this implementation does not fully comply with our theoretical concepts it is an effective manner to counter soon to be expected attacks, such as the man-in-the-browser attack (see Section 5.3). Subsequently, we present two branches that could be implemented on the long term and that fulfill our concept wholly.

6.2.1 Transaction information confirmation

Our first alternative is a migration of the current defensive strategy which has been described in Chapter 4. The key point of this migration is to be able to counter sophisticated phishing attacks that are prevalent or to be expected in the near future. Real-time man-in-the-middle attacks (see Section 3.1) and man-in-the-browser attacks (see Section 5.3) are the main examples of such attacks. These attacks exploit the weaknesses that are present in current two-factor and two-channel authentication schemes. As expressed in Section 5.3, the central problem with these implementations is that they depend on the correct and genuine reproduction of data by the customer's end-systems. However, as pointed out in our threat model in Section 2.4 it is not reasonable to trust these end-systems. Hence, we need to fit the aforementioned concept of *end-to-end security* in the current two-factor and two-channel authentication schemes.

The key issue in this migration is *transaction information confirmation*: we require the customer to confirm the transaction information (i.e. amount of the transaction and the account number of the beneficee) using the trusted part of the authentication scheme and independent of the customer's end-system. That means, it is confirmed either via the security calculator or via the second channel. In this way, the integrity of the transaction information does no longer depend on the correct and genuine reproduction of this information by the customer's end-systems. As a consequence, real-time man-in-the-middle attacks and man-in-the-browser attacks are rendered impossible.

Let us discuss how we can achieve transaction information confirmation in both a two-channel context and a two-factor context by adapting the transaction initiation protocols of these contexts:

In a two-channel context

In a two-channel context we can establish transaction information confirmation by creating a binding between the TAN code and the transaction information. The solution for this is very trivial: always send the transaction information along with the TAN code over the trusted channel (e.g. the SMS text messaging channel). Consequently, a customer can notice that a man-in-the-middle or man-in-the-browser attack is attempted when the transaction information confirmation received via the trusted channel diverges from the transaction information that he entered at his end-system. If this is the case, the customer should not confirm the transaction by entering the TAN code but instead call the bank and notify them that an attack is going on.

The message sequence chart in Figure 6.2 depicts the two-channel authentication scheme with transaction information confirmation:

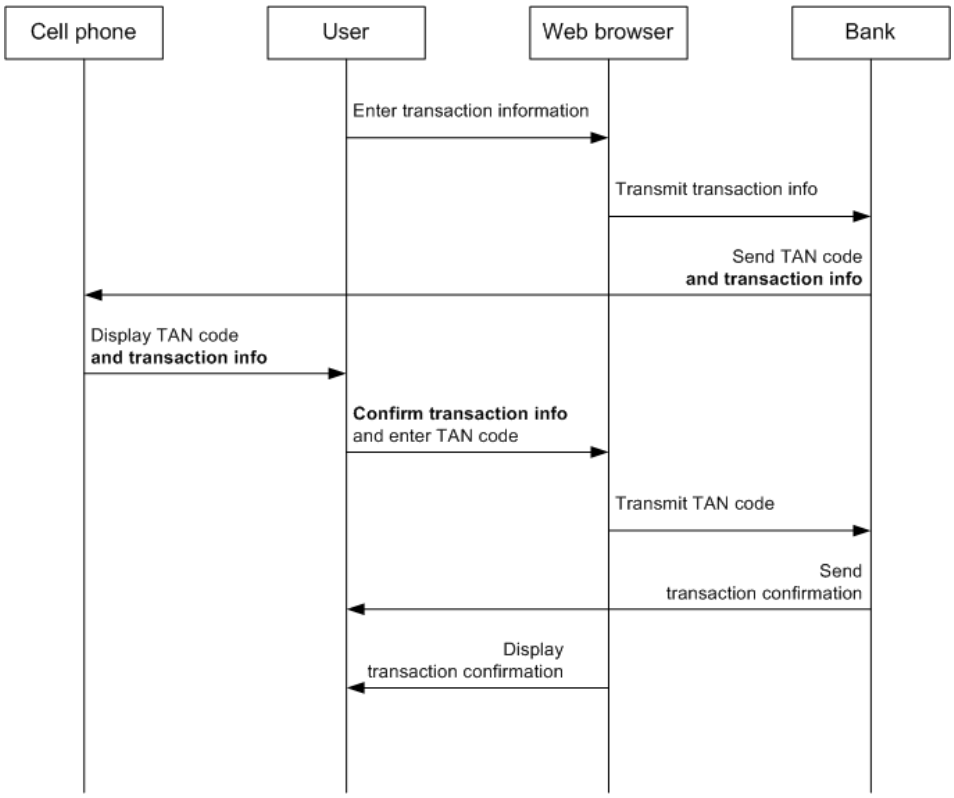

Figure 6.2: Message sequence chart of transaction information confirmation in a two-channel context

This scheme requires very little adjustments and could be rolled out on a very short term. Unfortunately, the major drawback of this scheme is that it requires a real-time trusted channel to deliver the transaction information confirmation. This means that this scheme is not applicable when TAN codes are transmitted in bulk mode via the postal service.

In a two-factor context

In a two-factor context we can establish transaction information confirmation by creating a binding between the challenge-response mechanism (in which the security calculator is involved) and the transaction information. We do so by letting the customer enter the transaction information in the security calculator (in addition of entering this information in the customer's end-system). The security calculator can then generate a digital signature of the transaction information which is sent along with the repsonse. Consequently, the bank can notice an attempted man-in-the-middle or man-in-the-browser attack when the signature does not match the transaction information that was received.

The message sequence chart in Figure 6.3 depicts the two-factor authentication scheme with transaction information authentication.

Essentially, this is the scheme Rabobank applies to transactions that involve extremely large amounts. However, in our opinion this is just security theatre: these large transactions are already flagged by the back-end systems because of regulatory compliance (see Section 2.2.3). In practice, phishers steal smaller amounts of money [13] which helps them to avoid detection by back-end systems. Moreover, widespread man-in-the-browser malware would be the ideal way for an attacker to redirect lots of smaller transactions. Hence, we think it is wise to introduce this transaction information confirmation scheme for all transactions.

Reflection

Transaction information confirmation helps us to achieve that the integrity of the transaction information does no longer depend on the correct and genuine reproduction of this transaction information by the customers' end-systems. Consequently, it is a successful control to counter real-time man-in-the-middle attacks and man-in-the-browser attacks.

For the customer transaction information confirmation has a major benefit:

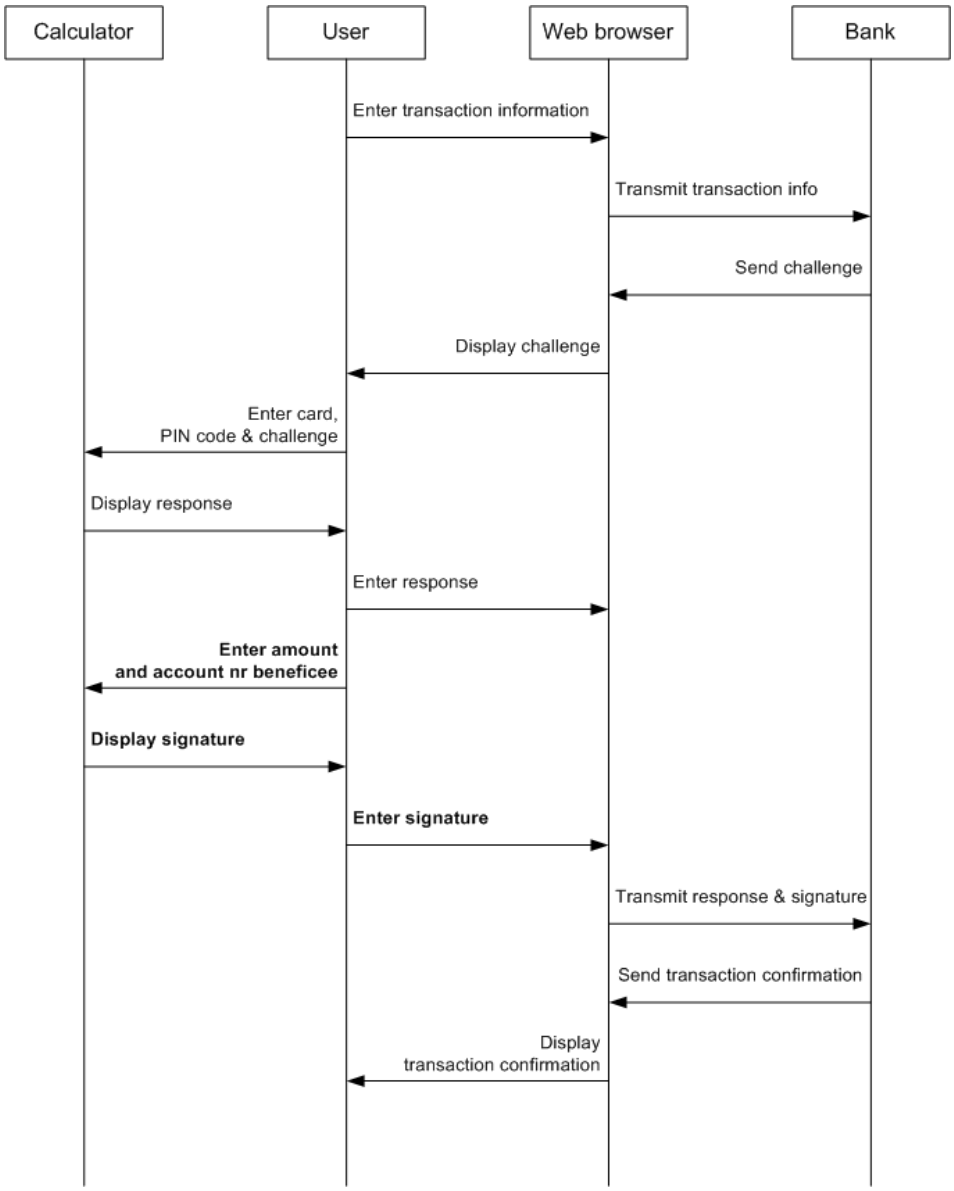

Figure 6.3: Message sequence chart of transaction information confirmation in a two-factor context

he can do secure transactions without the need to implement tight software security restrictions and mechanisms on his end-systems. In return, the customer should verify the information sent to him via a trusted channel (in a two-channel context) or enter the transaction information twice (in a two-factor context). We believe that transaction information confirmation is a major improvement: we think the increased security and decreased effort to maintain a secure end-systems outweigh the slightly increased effort to confirm the transaction information.

Moreover, we believe that transaction information confirmation is also favorable for the bank. It is a front-end control that is directly visible to the customer. As a consequence, we believe that it would not only improve security realization but also security perception by the customer. The latter is a great benefit with respect to the confidence in internet banking security (see also Section 2.1.2). Additionally, there is no need to shift the liability for internet banking security to the customer to maintain a secure end-system (see also the discussion in Section 4.3.3).

Altough transaction information confirmation is a quick fix to counter soon to be expected attacks, it does not fully meet our conceptual solution. For example, in a two-channel context a SMS text message that contains the transaction information can be easily neglected. Furthermore, transaction information only focusses on transactions as crucial internet banking activity. Altough it is probably the most important activity in internet banking it is not the only activity that should be protected. For example, changing one's correspondence address or making modifications to the shortlist of contacts is also crucial internet banking activity. We now provide two implementation branches of our conceptual solution that banks could roll out on the longer term and that do not suffer from these deficiencies.

6.2.2 Connected smart card reader

One way to implement our conceptual solution is using a secure connected smart card reader [26]. The smart card reader is connected to the customer's end-system via a standard interface, for example USB. The smart card reader also has a display, a numerical key-pad and a *sign* button. Since we ruled out physical access to such a device in our threat model (see Section 2.4) any requirements of tamper-proofness are out of the scope of our research.

Let us discuss the issues of this implementation:

- *Trusted computing base*
  In this implementation our TCB consists of the smart card reader and

the banking smart card.

- *End-to-end security*
  The smart card reader and the smart card are under direct control of the customer. The USB interface is only used to feed information to be signed to the smart card reader. In order to operate the signature application the user has to enter the PIN code that corresponds to his banking smart card.

- *Digital signatures*
  Either the smart card or the smart card reader must implement a digital signature algorithm, for example based on RSA [78]. We prefer to do this on the smart card so the signature algorithm can be easily updated by issueing a new smart card to a customer. Key management for this scheme is trivial: it can be based on keys already existing on the smart card. In the case of a compromised key only a new smart card has to be issued to a single customer. Furthermore, this has the advantage that all smart card readers can be identical, which saves costs and enables roaming. Secure timestamps are calculated by the smart card reader.

- *Non-negligible user-interface*
  All crucial internet banking activity requires a digital signature over the information involved in this activity. Before this can be done the customer is required to insert his smart card into the reader, enter his pin code and press the *sign* button. We stress that before pressing the *sign* button is important for the customer to carefully verify the information displayed on the smart card reader.

We are aware that this implementation has some similarities with FINREAD, the Financial transactional IC card reader [4]. However, FINREAD is a standard that is way more extensive than our proposal. For example, FINREAD allows a bank to remotely authenticate and update the software on the device. We believe that such functionality unnecessarily complicates the TCB, which does not support its security. Instead, we propose a simple smart card reader with security-critical software ran on the smart card.

We conclude this section by summarizing the considerations of this particular implementation:

- *Advantages*
  - Key management can be done based on the smart card.
  - A universal smart card reader allows for roaming.
  - Software updates can be rolled out by issueing new smart cards.

  - Because of current security calculators customers are familiar with using a separate device for signing purposes.

- *Disadvantages*

  - The smart card reader has to communicate with the customer's end-device which might cause interoperability issues.

6.2.3 Virtualization platform

Another way to implement our conceptual solution is to base it on a virtualization platform. We propose to use a type 1 hypervisor [15] that runs directly on hardware in order to avoid having to trust a huge operating system such as Windows or Linux. This hypervisor virtualizes two operating systems simultaneously. Namely, the customer's native operating system (e.g. Windows XP) and the bank's signing application. The hypervisor provides functionality for these operating systems to communicate.

Let us discuss the issues of this implementation:

- *Trusted computing base*
  In this implementation our TCB consists of the hypervisor and the signing application running on top of the hypervisor.

- *End-to-end security*
  The signing application has no network connection and can only be operated using hardware (e.g. mouse and keyboard) directly connected to the end-system. Hence, it is under direct control of the customer. The interface with the native operating system is only used to exchange information that is to be signed. In order to access the signature application the user has to enter a PIN code that was set up at installation of the signing application.

- *Digital signatures*
  The signing application must implement a digital signature algorithm, for example based on RSA [78]. Key management for this scheme is not trivial: the signing application must store the keys locally which complicates roaming. Moreover, depending on the digital signature algorithm being used a public key infrastructure has to be set up. Revoked keys require the user to install new keys in the signing application, which is a procedure that might enable new attack vectors. Secure timestamps can be calculated by the clock emulated by the hypervisor.

- *Non-negligible user-interface*
  All crucial internet banking activity requires a digital signature over

the information involved in this activity. Before this can be done the customer is required to switch to the signing application. When the customer recognizes the authenticity of the signing application by the background image he selected at installation he will enter his pin code. We stress that it is important for the customer to carefully verify the information displayed by the signing application. After that, the customer clicks the *sign* button.

Hopefully, in a few years time customers will have a standardized hypervisor integrated in their end-systems. This would allow a bank to develop a low-cost software-based signing application. However, it is unclear how this technology will develop and when it will be widely available in a standardized manner.

We round off this section by summarizing the considerations of this particular implementation:

- *Advantages*
  - The bank only has to develop a software signing application, which can be done at low costs.
  - The user does not have to switch to a seperate device for signing (however, this might cause disregard by the customer for the importance of the signature).
- *Disadvantages*
  - Key management is difficult.
  - Roaming is rendered difficult.
  - Future and standardization of hypervisors is unclear.

6.3 Key issues of this chapter

- A defensive solution that does not rely on the security of the customer's end-systems is desirable.
- Because of human behavior issues we require a solution that presents a simple and convenient interface to the customer and which can not be disregarded.
- Because of the trend to attack other internet banking activity that we observered, we should have a proper mechanism to protect all crucial internet banking activity and the data involved in this activity. Hence, we require a solution that allows customers to express their consciousness and the authenticity of their internet banking activity and to confirm the correctness of the information involved in this activity.
- Altogether, as a conceptual solution we propose a signing application running in a trusted computing base only available under direct physical control of the customer that requires a customer to digitally sign every crucial internet banking activity and the relevant information involved in this activity.
- One way to implement our conceptual solution is using a secure connected smart card reader.
- Another way to implement our conceptual solution is using a Type 1 hypervisor that emulates the native operating system of the customer and our signing application.
- On the short term these implementations will not be available. Nevertheless, we think that a mechanism to protect against man-in-the-browser attack is required on the short term. Therefore, we propose to adopt the protocols of the currently used security calculator and SMS-based TAN codes mechanisms to incorporate transaction information confirmation.

Chapter 7

Conclusions and recommendations

7.1 Conclusions

Let us answer our research questions that were posed in Section 1.3.2:

- *Q1: In what ways is it possible to structure the methodology of phishing techniques and defenses?*

  In Section 3.1 we devised a framework of seven steps that can be identified in a phishing hook. We also showed cohesion amongst these steps by which we divided phishing attacks in the lure, the hook and the catch. This disctinction follows the principle of separation of concerns which allows attackers to specialize. As a result, a phishing supply chain exists which is supported by illicit online market places. Currently, we identified the three most prevalent phishing attacks to be dragnet phishing, real-time man-in-the-middle phishing and malware based phishing.

  In phishing defense we can distinguish between front-end (see Section 4.1) and back-end controls (see Section 4.2). The former are directly visible to the customer while the latter are under direct control of the bank and mostly invisble to the customer. Prominent front-end controls are the authentication mechanisms. Here we can distinguish two-factor and two-channel authentication schemes. Moreover, front-end security relies on software security mechanisms that protect the security of the customer's end-systems. As back-end controls transaction anomaly detection and takedowns are most popular.

  The defensive strategy should be based on a proper threat model.

Unfortunately, an applicable threat model was not publicly available. Therefore we applied modelling techniques developed by Microsoft to construct a threat model ourselves (see Section 2.4). One of the main results from this threat model is that we identified tampering with the customer's end-systems a high-risk threat.

- *Q2: How will phishing attacks presumably evolve in the future?*

  In Section 5.1 we concluded that the evolution of phishing attacks will take place especially in the lure and hook of a phishing attack. We think that the developments of future attacks will be bipartite. On the one hand we will see the employment of new technology (broadening) and on the other hand we will see more sophisticated exploitation of current technologies (deepening). This is a direct result of the arms race between phishers and banks: when defense is raised phishers seek for new targets and techniques (broadening) and for ways to enhance current attacks in order to circumvent the raised security (deepening).

  In the lure we expect to see new methods for contacting victims (broadening). Vishing, the use of Voice over IP technology for phishing lures, is one example of such attacks. We showed that the required technology for vishing is publicly available at low costs. Moreover, we expect spear phishing (deepening) to develop. In spear phishing only a smaller number of interesting victims is selected in order to achieve a higher return on investment and in order to create more stealthy phishing attacks.

  In the hook of phishing attacks we expect to see sophisticated semantic attacks. We expect the current malware-based phishing attacks to evolve into man-in-the-browser attacks (deepening), which are attacks that redirect transaction initiations using a malware-infeced web browser. In Section 5.3.1 we demonstrated the practical feasibility of such an attack. Man-in-the-browser attacks can be realized at low costs and with moderate skills. Moreover, we expect future phishing attacks to target other internet banking activity than just the transaction initiation activity (broadening). In Section 5.3.2 we documented an attack that successfully exploits the transaction request procedure. Probably, we are the first to report on this attack.

- *Q3: To what extent will currently deployed anti-phishing mechanisms protect against future attacks?*

  A major pillar in phishing defense are the two-factor and two-channel authentication schemes employed by Dutch banks. These schemes suc-

cessfully rendered dragnet phishing infeasible. However, this moved phishers to attack other internet banking activity than just the login procedure. Namely, both two-factor and two-channel authentication schemes lack a proper mechanism to protect the integrity of the data involved in internet banking activity. Hence, in order to ensure the integrity of the data transmitted to the bank a secure path from the customer to the bank is required. This leads to a focus on software security mechanisms such as firewalls and virus scanners that protect the customer's end systems and to a focus on SSL / TLS for the protection of the connection to the bank. Unfortunately, these defensive controls have serious deficiencies (see Section 4.3). As a consequence we see that the transaction initiation procedure is attacked by phishing attacks such as real-time man-in-the-middle phishing and man-in-the-browser attacks. In conclusion, the currently deployed defensive strategy fails to adequately target the sophisticated phishing attacks that we expect to arouse in the near future.

- *Q4: Which defensive techniques are required to protect against the future prospects of phishing attacks?*

  In Section 4.1 we have seen that protecting the security of the customer's end-systems is infeasible because of the lack of proper controls and the nature of human behavior. As a consequence, we require a defensive solution that does not rely on the security of these end-systems. Moreover, because of this human behavior we require a solution that presents a simple and convenient interface to the customer and which can not be disregarded. Finally, because of the trend to attack other internet banking activity that was observered in answering Q3, we should have a proper mechanism to protect all crucial internet banking activity and the data involved in this activity. Hence, we require a solution that allows customers to express their consciousness and the authenticity of their internet banking activity and to confirm the correctness of the information involved in this activity.

  In Chapter 6 we present a solution: we propose a signing application running in a trusted computing base only available under direct physical control of the customer that requires a customer to digitally sign every crucial internet banking activity and the relevant information involved in this activity.

  On the long term, such a solution could be implemented as a secure smart-card reader or using a Type 1 hypervisor. On the short term such a solution will not be available. In the meanwhile we think it is wise to adapt current two-factor and two-channel authentication

schemes to adopt transaction information confirmation (see Section 6.2.1). This will render man-in-the-browser attacks infeasible.

7.2 Recommendations

Based on our conclusions we make the following three recommendations to internet banking services:

- A defensive strategy should not rely on a trusted path from the customer to the bank. This leads to liability issues, it is out of control of the internet banking service and it is inherently insecure since software security mechanisms have serious deficiencies and security indicators of SSL / TLS are easily ignored.
- Man-in-the-browser attacks are extremely feasible. A proper defensive control is required in order to avoid serious attacks, loss of customers' trust in internet banking security and a media smear campaign. We advise to adapt the current two-factor and two-channel authentication schemes to incorpate transaction information confirmation. For two-factor schemes this means that security calculators should require a digital signature over the transaction information (account number and amount). For two-channel schemes this means that when sending the TAN code to the customer over SMS text messaging a confirmation of the transaction information should be sent along.
- On the longer term we expect that other digital payment activity will also be the victim of attacks, as was demonstrated by the man-in-the-mailclient attacks discussed in Section 5.3.2. We advise internet banking services to seriously research these problems before attacks are carried out in the wild. A control that protects all crucial internet banking activity and the information involved in this activity is required. For example, look into the possibilities of connected smart card readers or emulation using a hypervisor, as was dicussed in Chapter 6.

Bibliography

[1] Anti-Phishing Working Group. *Proposed Solutions to Address the Threat of Email Spoofing Scams*, http://www.antiphishing.org, December 2003.

[2] Basel Committee on Banking Supervision. *Risk Management Principles for Electronic Banking*, http://www.bis.org/publ/bcbs98.pdf?noframes=1, July 2003.

[3] CA / Browser Forum. *Guidelines for the issuance and management of extended validation certificates*, Version 1.0, June 2007.

[4] CEN - European Committee for Standardization. *FINREAD specifications*, CWA 14174, http://www.cen.eu/cenorm/sectors/sectors/isss/cwa/finread.asp, 2004.

[5] Centraal Bureau voor de Statistiek. *De digitale economie 2006*, http://www.cbs.nl/NR/rdonlyres/D2E9E66D-D6A1-4438-83C4-541DC6D92B30/0/2006p34pub.pdf, 2006.

[6] Consumentenbond. *'Internetbankieren is veilig'*, Consumentengids, January 2008.

[7] De Nederlandsche Bank. *Internetbankieren nu en in de toekomst*, DNB / Kwartaalbericht, http://www.dnb.nl/dnb/home/file/Kwartaalbericht\%20compleet_tcm46-156016.pdf, June 2007.

[8] De Nederlandsche Bank. *Regeling organisatie en beheersing*, 2001.

[9] Department of Defense, *Trusted computer system evaluation criteria*, DoD 5200.28-STD, 1985. In the glossary under entry Trusted Computing Base (TCB).

[10] Federal Financial Institutions Examination Council. Authentication in an Internet Banking Environment, http://www.ffiec.gov/pdf/authentication_guidance.pdf, 2005.

[11] Federal Financial Institutions Examination Council. *IT Examination Handbook*, http://www.ffiec.gov/ffiecinfobase/booklets/information_security/information_security.pdf, July 2006.

[12] Federal Trade Commission's Division of Marketing Practices. *Email Address Harvesting and the Effectiveness of Anti-Spam Filters*, http://www.ftc.gov/opa/2005/11/spamharvest.pdf, November 2005.

[13] Gartner Inc. *Gartner Says Number of Phishing E-Mails Sent to U.S. Adults Nearly Doubles in Just Two Years*, Press Release, 9 November 2006. http://www.gartner.com/it/page.jsp?id=498245

[14] Govcert. *Trendrapport 2007*, http://www.govcert.nl/download.html?f=84, 2007.

[15] IBM Corporation. *IBM Systems Virtualization*, Version 2 Release 1, http://publib.boulder.ibm.com, 2005.

[16] Ministerie van Justitie, Directoriaat-Generaal Rechtshandhaving. *Kamervragen over phishing*, nr. 2040518750, Dutch, October 2005.

[17] Millward Brown / Centrum. *Perceptie en urgentie veiligheid op het internet*, http://www.xs4all.nl/nieuws/pdf/veiligheidsonderzoek.pdf, 2005.

[18] RSA Security Inc. *Fighting Emerging Threats: How To Combat Man-In-The-Middle And Trojan Attacks*, 2007.

[19] RSA Security Inc. *Fighting The Enemy: Making Sense of the Growing Crimeware Threat*, 2006.

[20] RSA Security Inc. *Phishing Special Report: What We Can Expect For 2007*, January 2007.

[21] SecuritySpace. *Secure Server Survey*, http://www.securityspace.com/s_survey/sdata/200710/certca.html, November 2007.

[22] World Wide Web consortium. *Document Object Model (DOM) Level 1 Specification Version 1.0*, http://www.w3.org, 1 October, 1998.

[23] C. Abad. *The economy of phishing: A survey of the operations of the phishing market*, Cloudmark, September 2005.

[24] B. Adida, S. Hohenberger and R. L. Rivest. *Fighting Phishing Attacks: A Lightweight Trust Architecture for Detecting Spoofed Emails*, http://people.csail.mit.edu/rivest/AdidaHohenbergerRivest-FightingPhishingAttacks.pdf, 2005.

[25] B. Adida, S. Hohenberger and R. L. Rivest. *Seperable Identity-Based Ring Signatures: Theoretical Foundations For Fighting Phishing Attacks*, presented at the DIMACS Workshop on Theft in E-Commerce, Piscataway, New Jersey February 2005.

[26] A. Allan, J. Heiser, A. Litan, A. Newton and R. Wagner. *State of the Art for Online Consumer Authentication*, Gartner, May 2006.

[27] A. Alsaid and C. J. Mitchell. *Preventing Phishing Attacks Using Trusted Computing Technology*, in Proceedings of INC 2006, Sixth International Network Conference, Plymouth, UK, pp.221-228, July 2006.

[28] M. Baaijens. *Prepare for VoIP Spam*, Master Thesis, November 2007.

[29] D. Bizeul. *Russian Business Network Study*, `http://www.bizeul.org/files/RBN_study.pdf`, November 2007.

[30] D. Barroso. *Botnets - The Silent Threat*, European Network and Information Security Agency (ENISA), November 2007.

[31] K. B. Bignell. *Authentication in an Internet Banking Environment; Towards Developing a Strategy for Fraud Detection*, International Conference on Internet Surveillance and Protection, 2006.

[32] M. Christodorescu and S. Jha. *Testing Malware Detectors*, Published in the Proceedings of the ACM SIGSOFT International Symposium on Software Testing and Analysis (ISSTA'04), July 2004.

[33] R. Clayton. *A Chat at the Old Phishin' Hole*, lecture notes in computer science number 3570, pages 88, Springer-Verlag, 2005.

[34] R. Clayton. *Who'd phish from the summit of Kilimanjaro?*, ISBN 978-3-540-26656-3, pages 91-92, 2005.

[35] L. Cranor, S. Egelman, J. Hong and Y. Zhang. *Phinding Phish: An Evaluation of Anti-Phishing Toolbars*, CyLab Technical Report CMU-CyLab-06-018, November 2006.

[36] J. Dapeng. *Personal Firewall Usability-A Survey*, TKK T-110.5290 Seminar on Network Security, 2007.

[37] P. Dasgupta, K. Chatha, and S. K. S. Gupta. *Personal Authenticators: Identity Assurance under the Viral Threat Model*, draft, 2006.

[38] L. Delpha and M. Rashid. *Smartphone Security Issues*, Black Hat Briefings Europe, May 2004.

[39] R. Dhamija, J. D. Tygar and M. Hearst. *Why Phishing Works*, in the Proceedings of the Conference on Human Factors in Computing Systems (CHI2006), April 2006.

[40] T. Dierks and E. Rescorla. *The Transport Layer Security (TLS) Protocol Version 1.1*, RFC 4346, April 2006.

[41] R. Dingledine, N. Mathewson and P. Syverson. *Tor: The Second-Generation Onion Router*, in proceedings of the 13th USENIX Security Symposium, pp. 303-320, 2004.

[42] A. Emigh. *Online Identity Theft: Phishing Technology, Chokepoints and Countermeasures*, Identity Theft Technology Council, `http://www.antiphishing.org/Phishing-dhs-report.pdf`, October 2005.

[43] I. Fette, N. Sadeh and A. Tomasic. *Learning to Detect Phishing Emails*, Carnegie Mellon Cyber Laboratory Technical Report CMU-CyLab-06-012, June 2006.

[44] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach and T. Berners-Lee. *Hypertext Transfer Protocol – HTTP/1.1*, Request for Comments 2616, June 1999.

[45] P. Finn and M. Jakobsson. *Designing and Conducting Phishing Experiments*, preprint, to appear in IEEE Technology and Society Magazine, Special Issue on Usability and Security, 2007.

[46] J. Franklin, V. Paxson, A. Perrig and S. Savage. *An Inquiry into the Nature and Causes of the Wealth of Internet Miscreants*, in Proceedings of the 14th ACM conference on Computer and communications security, pages 375-388, 2007.

[47] G. R. Gordon, D. J. Rebovich, K. Choo and J. Gordon. *Identity Fraud Trends and Patterns: Building a Data-Based Foundation for Proactive Enforcement*, Grant No. 2006-DD-BX-K086, October 2007.

[48] M. G. Gouda, A. X. Liu, L. M. Leung and M. A. Alam. *SPP: An anti-phishing single password protocol*, in Computer Networks: The International Journal of Computer and Telecommunications Networking Volume 51, Issue 13 (September 2007), Pages 3715-3726, March 2007.

[49] P. Gühring. *Concepts against Man-in-the-Browser Attacks*, Financial Cryptography, FC++ number 3, 2007.

[50] P. Gutmann. *Phishing Tips and Techniques*, presentation at the University of Cambridge Computer Laboratory, May 2007.

[51] A. Hallawell and A. Litan. *Brand-Monitoring and Anti-phishing Services Intersect Several Security Markets*, Gartner, September 2007.

[52] M. Howard and D. LeBlanc. *Writing Secure Code*, Microsoft Press, 2002.

[53] C. Jackson, D.R. Simon, D.S. Tan and Adam Barth. *An Evaluation of Extended Validation and Picture-in-Picture Phishing Attacks*, In Proceedings of Usable Security (USEC '07) Workshop, 2007.

[54] T. Jagatic, N. Johnson, M. Jakobsson and F. Menczer. *Social Phishing*, in Communications of the ACM Volume 50, Issue 10 (October 2007), Pages 94 - 100, December 2005.

[55] M. Jakobsson. *Modeling and Preventing Phishing Attacks*, in Phishing Panel of Finfancial Cryptography, 2005.

[56] M. Jakobsson. *Phishing and Countermeasures: Understanding the Increasing Problem of Electronic Identity Theft*, Wiley, ISBN: 978-0-471-78245-2, 2007.

[57] M. Jakobsson and J. Ratkiewicz. *Designing Ethical Phishing Experiments: A study of (ROT13) rOnl query features*, In Proceedings of the 15th International Conference on World Wide Web, 2006.

[58] M. Jakobsson and A. Young. *Distributed Phishing Attacks*, `http://eprint.iacr.org/2005/091.pdf`, 2005.

[59] L. James. *Phishing Exposed*, Syngress, ISBN: 978-1-597-49030-6, November 2005.

[60] J. Klensin. *Simple Mail Transfer Protocol*, RFC 2821, April 2001.

[61] B. Lampson, M. Abadi, M. Burrows and E. Wobber. *Authentication in Distributed Systems: Theory and Practice*, ACM Transactions on Computer Systems, on page 6, 1992.

[62] A. Litan. *HSBC Bank Brasil Turns to Back-End Fraud Detection to Curb Cybercrime*, Gartner, June 2006.

[63] A. Litan. *Phishing Attacks Leapfrog Despite Attempts to Stop Them*, Gartner, November 2006.

[64] S. Mauw and M. Oostdijk. *Foundations of Attack Trees*, presented at Eighth Annual International Conference on Information Security and Cryptology, 2006.

[65] M. C. McChesney. *Banking in cyberspace: an investment in itself*, IEEE Spectrum, February 1997.

[66] A. McCullagh. *Non-Repudiation in the Digital Environment*, First Monday, volume 5, number 8, August 2000.

[67] J.D. Meier, A. Mackman, M. Dunner, S. Vasireddy, R. Escamilla and A. Murukan. *Improving Web Application Security: Threats and Countermeasures Roadmap*, Microsoft Corporation, June 2003.

[68] T. Moore and R. Clayton. *An Empirical Analysis of the Current State of Phishing Attack and Defence*, In Proceedings of the 2007 Workshop on The Economics of Information Security (WEIS2007), `http://www.cl.cam.ac.uk/~rnc1/weis07-phishing.pdf`, 2007.

[69] P. Mutton. *PayPal Security Flaw allows Identity Theft*, Netcraft, June 2006.

[70] G. Ollman. *The Phishing Guide: Understanding & Preventing Phishing Attacks*, NGSSoftware Insight Security Research, 2004.

[71] B. Parno, C. Kuo and A. Perrig. *Phoolproof Phishing Prevention*, CMU-CyLab-05-003,`http://sparrow.ece.cmu.edu/~adrian/projects/phishing.pdf`, March 2006.

[72] N. Provos. *A Virtual Honeypot Framework*, In Proceedings of the 13th USENIX Security Symposium, August 2004.

[73] N. Provos, P. Mavrommatis, M.A. Rajab and F. Monrose. *All Your iFRAMEs Point to Us*, Google technical report provos-2008a, 2008.

[74] N. Provos, D. McNamee, P. Mavrommatis, K. Wang and N. Modadugu. *The Ghost In The Browser - Analysis of Web-based Malware*, Google Inc., 2007.

[75] J. Quirke. *Security in the GSM system*, AusMobile, May 2004.

[76] B. Reijnen. *Internetbankieren: magneet voor criminelen*, Elsevier (in Dutch), November 2007.

[77] E. Rescorla. *SSL and TLS: Designing and Building Secure Systems*, ISBN 978-0201615982, Addison-Wesley Professional, 2000.

[78] R. Rivest, A. Shamir, and L. Adleman. *A Method for Obtaining Digital Signatures and Public-Key Cryptosystems*, Communications of the ACM, Vol.21, Nr.2, 1978, S.120-126.

[79] P. Robichaux and D. L. Ganger. *Gone Phishing: Evaluating Anti-Phishing Tools for Windows*, 3Sharp LLC, `http://www.3sharp.com/projects/antiphishing/gone-phishing.pdf`, September 2006.

[80] B. Ross, C. Jackson, N. Miyake, D. Boneh and J. C. Mitchell. *Stronger Password Authentication Using Browser Extensions*, Proceedings of the 14th Usenix Security Symposium, 2005.

[81] S. E. Schechter, R. Dhamija, A. Ozment and I. Fischer. The Emperor's New Security Indicators - An evaluation of website authentication and the effect of role playing on usability studies, The 2007 IEEE Symposium on Security and Privacy, May 2007.

[82] B. Schneier. *Attack trees: Modeling security threats*, Dr. Dobb's journal, December 1999.

[83] B. Schneier. *Semantic Attacks: The Third Wave of Network Attacks*, Crypto-Gram Newsletter, 2000.

[84] F. Swiderski and W. Snyder. *Threat Modeling*, Microsoft Professional, ISBN 978-0735619913, July 2004.

[85] G. Tally, R. Thomas and T. van Vleck. *Anti-Phishing: Best Practices for Institutions and Consumers*, McAfee, March 2004.

[86] T. G. Tan. *Phishing Redefined - Preventing Man-in-the-Middle Attacks for Web-based Transactions*, Draft, March 2005.

[87] K. Thompson. *Reflections on trusting trust*, in Communications of the ACM archive Volume 27 , Issue 8, August 1984.

[88] J. Verdurmen. *Firefox extension security*, Bachelor thesis, January 2008.

[89] L. Wang and P. Dasgupta. *Kernel and Application Integrity Assurance: Ensuring Freedom from Rootkits and Malware in a Computer System*, 21st International Conference on Advanced Information Networking and Applications Workshops, 2007, Volume 1, pages 583-589, 2007.

[90] Z. Ye, S. Smith and D. Anthony. *Trusted Paths for Browsers*, in Proceedings of the 11th USENIX Security Symposium, Pages: 263 - 279, 2002.

[91] A. Young and M. Yung. *Malicious Cryptography: Exposing Cryptovirology*, Section 8.2, ISBN 978-0764549755, February 2004.