Security, Privacy and Efficiency of Internet Banking Transactions†

C. Ronchi, A. Khodjanov, M. Mahkamov, S. Zakhidov
EISST Development Laboratory
edl@eisst.com

Abstract

The last decade has witnessed the emergence of a plethora of approaches for securing financial transactions over the Internet. During the same period, attacks have matured from isolated exploits to an organized e-criminal industry. In the midst of this evolution stood the End User, whose instances have often been neglected under the assumption that refunding financial losses is all that mattered. This paper analyzes the existing deployments of Internet banking services from the perspective of the End User, whose main goal is completing the online transaction. The sole use on the client side of so-called "trusted" hardware devices will be discussed and shown to fall short of the requirements for truly secure Internet banking.

Evidence will be provided in support of the need to protect the client components using connected devices and applying software hardening techniques to lower the hacking ROI and help rebalance forces in the fight against cyber criminals. A new metric for gauging the effectiveness of security software will be described and applied to measure the practical security of existing Internet banking systems. Finally, a number of guidelines will be provided for assuring that reasonable care is exercised in the design and deployment of Internet banking systems.

1. Introduction

It is safe to assume that the average reader of this paper understands the root causes of the observed insecurity of Internet transactions. Nonetheless, it may be useful to pause on the role that End Users play in this scenario. The relevance of this question might seem obvious, inasmuch the ultimate goal of Internet banking should be to support the Banks' business model, *i.e.* to encourage the use of the online service delivery model. In other words, any proper solution to the security issues concerning Internet banking cannot solely impose technology over the ability to uphold adoption by End Users. Indeed, it is a quite common experience for security professionals to observe theoretically sound security measures totally thwarted by careless, unaware or impatient End Users. Unfortunately this observation has led to the conclusion that the deployment of End User-proof systems will provide the long sought panacea. Stated differently, the prevailing opinion regarding the proper solution to the insecurity dilemma of Internet banking transactions seems to be aptly summarized by the following statement:

"*The End User's PC and the End User's ability to make the proper decisions cannot be trusted. Thus, let's eradicate the security of the Internet transaction from both the host PC resources and from the End User's ability to make the right judgment call.*"

At a first reading, this statement looks reasonable and seems to indicate a feasible resolution. After all, isolating the security-critical elements of the Internet transaction from the targets of malicious attacks (*i.e.* the PC and the End User), should also remove the enabling factors of online fraud and settle the issue.

The rest of this paper is dedicated to the analysis of this latter conclusion. We will show that the sustainability of the Internet business model requires more than just a "provably" secure fraud prevention system and that the viability of Internet banking transactions is strictly connected to the legitimate desire of the End Users to easily, efficiently and confidentially use their PC to transact online.

2. Principles and practice of Internet banking

It is estimated that by the end of year 2011 about 2.5 billion people will be using an Internet browser for recreational or business activities. The general consensus is that about half of this vast population of users can be considered as *soft-targets* of attacks from malware, while over a third of all Internet users have their PC already infected and possibly enrolled in botnets. Although there are no official figures, it is reasonable to assume that the same conclusions apply to the End Users of Internet banking services, *i.e.* the PCs and browsers employed to transact are either already under the control of e-criminals or can be easily infected. Thus, short of asking End Users of installing and employing a separate and isolated

(†) This article will appear in the proceedings of the World Congress on Internet Security (WorldCIS-2011) February 21-23, 2011, London, UK workstation only for Internet banking, one must confront the quandary of hinging the delivery of Internet banking services on the use of personal computers controlled by fraudsters.

Students of information security courses are taught that "*all the information security controls and safeguards, and all the threats, vulnerabilities, and security processes are subject to the C.I.A. yardstick*", where the acronym stands for the three Infosec tenets, namely confidentiality, integrity and availability. It is fair to expect that Bank and security professionals must strive to apply this guideline also to the design, adoption and deployment of Internet banking systems.

It is also customary to classify all the numerous possible attacks against End User computers/devices and online service delivery models in three major categories: physical cracking, social engineering and malicious software (or malware). As we will see in more detail in the next section, different attack vectors must be active in order to enable an attack from one of the above three categories. In general, it is impossible to craft a successful attack against an Internet banking system using one single attack vector. Indeed, the most powerful attacks are actually the results of using a combination of attack vectors from the above attack categories. This provides the rationale for advocating a holistic approach to security, whereby one should strive to deactivate the largest number of attack vectors across as many attack categories as possible.

The above considerations may help to articulate the terms of an acceptable deontology for Internet banking. In particular, and as preposterous it may sound, we'd like to question whether the ultimate and guiding principle for the design of Internet banking systems should solely be the avoidance of financial frauds. More precisely, we believe that it is arguable that Banks and financial institutions should also be concerned with the "collateral damages" inflicted to End Users by inducing them to use an inherently insecure computing system to access their online services. The plagues of identity theft, personal data disclosure and privacy loss are clearly associated to the fact that the main target of the e-criminals are the computers employed by End Users for Internet banking. Under this light, we believe that it is objectionable to deploy protection technologies that do not also safeguard the End User's privacy and that focus only on providing means for stopping potentially fraudulent transactions.

For the sake of clarity, let us restate and extend the above conclusion with the help of a *gedanken* experiment. To this end, assume that it is possible to install on the End User's computer a system which detects infallibly whether the Internet banking transaction has been compromised by an attack from any of the aforementioned categories and communicates this information to the End User. Imagine, however, that the system can accomplish such stunning task only *after* the End User has completed the entire transaction process on his computer. Now, can such an *after*-system be considered as a definitive and acceptable solution to the current Internet banking security dilemma? We believe that there are at least two arguments that should lead to a negative answer.

Firstly, from the End User's perspective, during an Internet banking session there is much more at stake than just the final result. In fact, for obvious authentication and procedural requirements, during the transaction the End User is asked either to enter or to access information regarding its standing with the Bank. This data is handled by the application running on the End User's computer (typically an Internet browser) and it is shown on the external display or entered using the keyboard. The deployed *after*-system doesn't mitigate the risks associated with making this information available to attackers, who can use it to craft new targeted malware, for social engineering attacks or even for blackmailing.

Secondly, the *after*-system cannot improve a key parameter required for high-quality Internet banking systems, *i.e.* the transaction *efficiency* (defined as the ratio of completed transactions over aborted transactions). In other words, from the perspective of the End User, the principal goal of any improved transaction validation method should be the ability to reduce the current level of exposure to online fraud while at the same time enabling a growing number of legitimate transactions. Under the realistic working assumption that the End User computer is easily prey of malware attacks, the *after*-system can at best stop a fraudulent transaction but it only indirectly supports the Banks' online business model, whereby End Users should become growingly confident in their ability to successfully and securely carry out banking transactions over the Internet. Improving the detection of fraudulent schemes without providing means for increasing the transaction efficiency will not serve the system's end purpose, ultimately frustrate End Users and raise serious questions on the sustainability of the online business model.

The above analysis clearly suggests that, based on the commonly accepted C.I.A. yardstick, the *after*-system cannot be considered to be secure. In fact, under attack (which is the only scenario of interest) it fails to satisfy two out of the three yardstick criteria, namely confidentiality and availability.

3. Provable versus practical security

A number of real world Internet banking systems operate similarly to the *after*-system described in the previous section. Such systems are all based on some form of out-of-band verification method and can be classified in two major categories depending on whether or not the transaction is verified using a device connected to the End User's computer. Examples of the latter methods include mTAN (*aka* SMS validation), optical TAN (*aka* flickering device) and various forms of TAN calculators with or without Smart Card readers. These latter can also function in connected mode, thereby allowing to receive the transaction data to be signed directly from the PC and to show it to the End User on an embedded display. Another example of a connected *after*-system is the Zone Trusted Information Channel device (*aka* ZTIC) which implements a trusted TLS protocol endpoint directly on the device instead of relying on that provided by the PC.

All the methods listed above share the same operating principles, which can be summarized as follows:

- Let the End User carry out the Internet banking transaction on his/her PC and using a standard Web browser;
- send and show the transaction data to the End User using a separate secure channel;
- ask the End User to verify and confirm the validity of the transaction details using the external device.

The leading rationale here is to deploy a transaction validation system which is *provably* secure (to use a common terminology used for such systems) whenever it is used *properly* by the End User. The analysis of the previous section, however, suggests a quite different and somewhat paradoxical conclusion, *i.e.* that an *after*-system may be both practically insecure and practically inefficient, since it hinges on the End User's ability to take the proper decision while using a compromised software and PC. Under this light, it is inaccurate to refer to such systems as provably secure, since a better representation is provided by the term "provably *integral"*. Plenty of examples can be drawn from the security practice whereby hackers are capable of inducing individuals into neutralizing or misusing even very sophisticated protection techniques. As we will discuss in more detail in the next section, social engineering attacks are highly potentiated by the ability to obtain sensitive information on the transaction and to control its key interface elements. This synergy may lead to the practical failure of otherwise theoretically sound security measures.

This conclusion is supported by a field study [2] carried out using a test e-banking scenario, wherein End Users were asked to complete Internet banking transactions in a controlled environment using a standard Web browser and a simulated *after*-system. The End Users were instructed to login using a strong authentication method (*e.g.* a fingerprint scan) and then to carefully check the transaction details before authorizing any payments. In particular, the account number of the beneficiary was displayed and showed to the End Users on a separate screen before the authorization step could be completed. The End Users could then authorize or abort the transaction by clicking on two separate buttons (red, not OK – green, OK). During the sessions, the End Users were presented with several real-life legitimate and fraudulent scenarios, including maintenance and support messages from the Bank personnel, phishing scams and more sophisticated attacks using social engineering and interface manipulation techniques.

Results showed that under attack the practical security of transaction validation methods based solely on the use of an *after*-system is only marginally better than the security attainable using less sophisticated (and less costly) methods. In particular, the ability to manipulate the user interface of the Web browser during the transaction procedure enabled crafting a number of very effective attack schemes for inducing the End User to authorizing fraudulent money transfers. Furthermore, whenever the End User was alert enough to understand the danger and thwart the attack by clicking on the red button, he/she voiced a strong disillusion with respect to the trustworthiness of the overall business process sponsored by the Bank.

In conclusion, the key factors underlying the above observations (probably already well known to a large segment of security professionals) can be summarized as follows:

- the End User's acceptance of a new transaction process based on the use of an external device requires explaining the unique security benefits of such improved method. In order to surmount the acceptance hurdle, the End User is induced to place uncritical trust in the effectiveness of the new security process;
- during the online transaction, the main focus of the End User's experience is captured by the PC interface elements and delivered via the Web browser GUI components. These latter have a stronger influence on directing the End User decision process with respect to the data displayed on the external device;
- the End User doesn't critically analyze the Bank's Web site appearance, structure or any interactive messages displayed during the session. Once the End User is logged-in and operates within the e-banking environment, he/she believes in the integrity of the Web experience.

4. Attack vector activity analysis

It is not uncommon to hear the opinion that measuring security is about using common sense since quantitative measures of security are too complex to define given the dynamic nature of threats. Indeed, some even believe that security is a metric-adverse process, which cannot be reduced to technology considerations. Of course, unstructured attempts at measuring a system's general security are bound to fail for many good reasons. One indisputable fact, however, is that the least vulnerable a system the lower can be considered its security risk. In the case of software, the problem is mainly the determination of what constitutes vulnerability and what to make of it, since one can observe that annually over 50% of the most prevalent and critical vulnerabilities are replaced by new ones. A solution to this conundrum may come from focusing on attack vectors rather than on vulnerabilities, whereby the latter are exploited to activate a constant set of attacks vectors which in turn are required to support malware. In our terminology, an attack vector is defined as an elemental constituent of malware, necessary to enable at least one essential component of a malicious attack procedure. In this usage, new vulnerabilities can replace old ineffective ones to enable the *same* attack vectors.

A possible metric for gauging the practical effectiveness of security software can be based on an Attack Vector Activity Analysis (or AVAA) which hinges on a detailed taxonomy of the attack vectors exploitable by malware *after* the protections are deployed. The underlying assumption is that hacking will be more effective and economical (from a return on investment perspective) whenever a larger number of attack vectors are left active after the protection software has been deployed.

Notwithstanding its limited scope, the AVAA offers several practical benefits:

- it supports the principle that security must be holistic and enforced across all attack categories;
- it provides security experts with common starting grounds for more detailed threat analysis and risk determinations;
- it focuses the attention on attack vectors as the enabling constant components of malware;
- it helps the non-technical evaluation process of commercial solutions by providing a list of indicators correlated to the ability to protect against malware observed in the wild;
- it spurs the improvement of current protection methods by highlighting their strengths and weaknesses

At a first approach, the AVAA can be applied using simple and standard (often off-the-shelf) tools and methods to test for the activity of a pre-determined list of attack vectors. The chosen tools should be clearly disclosed and available to any party that wishes to replicate and validate the results of the analysis. Once an attack vector is found to be active, no determination should be made on the efforts required to bring it back to an active status. This is beyond the main scope of the AVAA, which primarily focuses on a high-level analysis of the attack vectors left active after a software protection system is deployed. Nonetheless, it is remarkable the wealth of information that can be obtained on the protection strength of a security software using even such a relatively coarse level of analysis.

The AVAA metric can of course be applied also to evaluate different approaches for securing online transactions. To this end, one can consider the following set of attack vectors grouped into four different categories particularly relevant to Internet banking scenarios:

TABLE 1. NETWORK ATTACK VECTORS

VECTOR	DESCRIPTION
DNS Spoofing **IP Rerouting (Transparent Proxy)**	Replacing of IP addresses in a DNS server or in a router to redirect the Victim to a malicious web resource.
URL Spoofing	Masking the malicious URL under a legitimate FQDN.
Eavesdropping	Intercepting and logging the network traffic
Content Spoofing	Intercepting and changing network traffic and web content

TABLE 2. INTERFACE ATTACK VECTORS

VECTOR	DESCRIPTION
Screen Capture	Ability to take an image of the screen of the PC while the transaction is in progress
Windows Overlay	Displaying different objects, such as buttons, notification messages and warning dialogs by overlaying these onto the legitimate application window
GUI Controls Manipulation	Manipulation of several GUI elements, such as buttons, window titles, menus, *etc*.
Keystroke Logging	Logging keyboard entries
Mouse Logging	Logging mouse events
Keyboard and Mouse Emulation	Generating keyboard and mouse events without requiring any physical action by the End User.

TABLE 3. PROCESS ATTACK VECTORS

VECTOR	DESCRIPTION
Dynamic Memory Read	Ability to access and analyze an application's process memory.
Dynamic Memory Patch	Ability to modify parts of an application's process memory.
Function Inject	Ability to inject a function into the address space of an application's process.
Dynamic Reverse Engineering	Ability to analyze the memory mapping of an application's executable.

TABLE 4. APPLICATION ATTACK VECTORS

VECTOR	DESCRIPTION
Component Manipulation	Modification of the application's components, such as add-ons, plug-ins, extensions, etc.
Static Reverse Engineering	Ability to reconstruct the internal logic of an application from its executable code.
Static Code Analysis	Ability to analyze the executable code of an application.
Static Code Patching	Ability to modify the executable code of an application.
Resource Patching	Ability to modify an application's resource files.

In the context of Internet banking, it is interesting to carry out the AVAA for the following five possible protection systems:

C1 – Disconnected device coupled with a standard Web browser (*e.g.* optical TAN, mTAN or TAN calculator)

C2 – Connected device with a standard Web browser (*e.g.* connected class 3 Smart Card readers)

C3 – Connected device with a TLS endpoint coupled to a standard Web browser (*e.g.* ZTIC) [3]

C4 – Secured Web browser (*e.g.* Rapport) [4]

C5 – Hardened Web browser (*e.g.* H-Browser) [5]

C6 – Connected device with on-board hardened Web browser (*e.g.* CLX.Sentinel.Display) [6]

The corresponding AVAA scores are shown in the table below, measured on a scale from 0 (no protection) to 5 (max protection):

TABLE 5. AVAA SCORES

	C1	**C2**	**C3**	**C4**	**C5**	**C6**
Network	0	0	5	4	5	5
Interface	0	0	0	2-3	5	5
Process	0	0	0	2-3	3-4	4
Application	0	0	0	3-4	3-4	5

As expected, since the above table focuses on the additional protection provided by security software, the AVAA scores for the C1, C2 and C3 solutions are not particularly meaningful since they simply indicate that these methods don't employ any security software client component. The gap between the scores of the C4 and C5 solutions reflects instead the lower effectiveness of using external software hardening versus internal (*i.e.* architectural) software hardening techniques [1]. Finally, the solution C6 corresponds to the most powerful combination available as of today, whereby a connected USB device with Smart Card, embedded keyboard and display, is also equipped with a multi-partitioned flash memory for storing and launching a hardened Web browser.

It is possible to carry out yet another comparison between the different solutions, but this time focusing more on business-relevant attributes, as shown in the table below:

TABLE 6. SOLUTION COMPARISON

	C1	**C2**	**C3**	**C4**	**C5**	**C6**
Privacy	0	0	0	3	5	5
Usability	2	3	3	5	5	4
Efficiency	2	2-3	3	3	4	5
Provability	yes	yes	yes	no	no	yes

Notwithstanding the fact that the hardware-only protection systems (C1-C3) can provide provable transaction integrity, they cannot protect the End User's privacy and are generally low in both usability and efficiency. The use of a hardware device allows increasing the efficiency of the C6 system above that of the software only solution C5, since one can use the flexible and powerful hardware features to implement very effective tamper-proof techniques to protect the on-board hardened browser.

5. Field Deployments

The current landscape of systems deployed to protect Internet banking is extremely composite and varied. By itself, this fact is quite surprising and can be understood by two possible explanations: either there is a lack of consensus on the best means for fighting cyber-criminals or the Banks withhold from investing resources to deploy the systems considered most effective at eradicating online fraud. As often happens, most likely the truth lies somewhere in the middle, whereby Banks consider a certain level of fraud as acceptable unless it severely impacts their business reputation and/or bottom line. Of course, the opinion of the End Users who will fall victims of such *acceptable collateral damage* are nowhere to be heard in the closed management meeting rooms where the decisions are made on whether to deploy an OTP device or a transaction signing system. Interestingly, online banking fraud is possibly one of very few phenomena with considerable social impact which still eludes proper reporting and full disclosure. It is arguable that a detailed discovery of the amount of online fraud suffered by each financial institution should be mandated by law and could provide a very strong incentive for Banks to "*do the right thing*".

What may be the *right* thing to do can vary somewhat from case to case. We believe, however, that there is sufficient evidence that points to what is the *best* thing that can be accomplished today to protect Internet banking sessions. In fact, the analysis of the previous sections clearly identifies this champion system to consist in a trusted client device equipped with a tamper proof storage for digital identities and with memory to store and launch a hardened application required to access the Internet banking services. The application should exploit to the maximum extent possible the hardware device to check and safeguard its integrity, as well as to prevent static patching and static analysis of the executable code and its resources. This can be accomplished by using obfuscation and byte-code virtualization techniques, as well as by storing critical components of the application inside of memory locations which are never mounted by the PC operating system and can be accessed only via firmware. The application can be made to execute inside of an isolated runtime environment, where all processes can be strictly monitored and policed.

It should be stressed that the solution outlined above (referred to as C6 in section 4) is based on mature technology components and is currently planned for deployment in spring 2011 at several Swiss banks. The feedback from early adopters, the overall security and measured transaction efficiency will be reported in a future paper.

6. Reasonable Care or Culpable Neglect?

In the previous sections we indirectly argued that the well-known saying "*security is like a chain*…" could be aptly complemented by the more prosaic "*security is like an onion*". Indeed, this comparison was shown to be particularly relevant to the practical domain of Internet banking inasmuch one should strive to decrease the hacking ROI by deploying layers of protections requiring specific efforts to be removed. The key determination, of course, is to identify the number and type of layers that can be reasonably deployed for Internet banking to still be considered viable and sustainable for all parties involved. Fortunately, this is not a Tantalus' task and can be accomplished by exercising an average degree of good will, professional due diligence and intellectual honesty. Lacking such predisposition may lead the Banks to neglect the risks to which End Users are exposed during Internet banking. For obvious marketing reasons, this also often requires the Bank to publicly overstate the security of their Internet banking system and to seek ways to limit its liability for the losses expected from online fraud.

One incentive for the Banks to exercise their best efforts in providing End Users with good and fair security measures may be provided by the recent upsurge of lawsuits from enraged End Users claiming that the financial institution didn't exercise reasonable care in protecting their Internet banking sessions [9, 10]. While the broader impact of these legal rulings is still unclear, one cannot help but feel disconcerted by the lack of leadership evidenced by the banking community as a whole in failing to agree to common and acceptable standards for the security of Internet banking systems. Such a concerted action would allow investing greater resources for R&D in banking security, ultimately benefiting all financial institutions and enabling them to solely compete on their core business values.

Since it is not wise to expect that such utopian turn of events will occur anytime soon, let us conclude by outlining a few guidelines for assuring that reasonable care is indeed exercised in the design and deployment of Internet banking systems.

Firstly, *safeguard the End User's privacy*, by providing means for preventing the capturing of personal data and the disclosure of sensitive information during the banking transaction.

Secondly, *lower the hacking ROI* by measuring the efforts required to offset the deployed protections from the perspectives of e-criminals. Make all efforts to limit the scalability of any successful attacks.

Thirdly, *think usability* and keep it as simple as possible. When in doubt, count and limit the number atomic operations required from the End User to complete a transaction. Analyze the transaction flow and minimize the occurrence of critical incidents.

Finally, *inform the End Users*. Always enforce a transparent and honest communication with the End Users on the residual risks associated with the use of Internet banking services. Strive to always exercise intellectual honesty when affirming the practical security of the deployed system.

7. Conclusions

The consistent and exponential growth in the volume, sophistication and pervasiveness of cyber attacks to Internet banking systems poses a very real question regarding the sustainability of the online business model. While attacks are primarily focused on the client's soft-components (PC, Internet browser, End User), the vast majority of the protection systems presently deployed by the Banks doesn't attempt to mitigate the risks directly associated with the endpoints and concentrates efforts on providing separate means for remotely validating the transaction details.

The analysis presented in this paper suggests that it is not possible to ignore the practical and social consequences of End Users transacting using a PC infected by financial malware and a general purpose non-hardened Internet browser. The rationale for this conclusion is found in the number of active endpoint attack vectors which enable e-criminals to craft powerful attacks that can manipulate the End User behavior, ultimately leading to either aborted or fraudulent transactions. Notwithstanding the general belief, the End Users who fall victim of cybercrime suffer damages which are not merely limited to financial losses, but embrace their privacy sphere and can impact their security for a long time after the fraud has been perpetrated and possibly reimbursed.

The present analysis also suggests that a properly designed transaction system comprised of *both* hardware and software components can lead to greatly improved levels of efficiency and security for Internet banking, while still safeguarding the overall usability and the End User's privacy. Indeed, a hardware device connected to the PC can be shown to be essential for providing effective tamper-proofing and a secure platform for storing and launching hardened software applications. The goal is to equip End Users with a usable system which will substantially complicate the hackers' tasks, lowering their ROI and helping to rebalance forces in the fight against the cyber criminals.

Finally, in this paper we have also somewhat naïvely attempted to spur the development of a higher deontology for Internet banking. We believe Banks should recognize the concerns and the needs of their customers beyond mere financial considerations (sic!), and exercise the competence, care and honesty that their vital social and public function requires also when it comes to developing, choosing and deploying technology solutions.

8. References

[1] C.Ronchi, S.Zakhidov, “A Road Map Towards a Practically Secure and Portable e-Banking Platform”, Technical Report ECS2008-02, EISST Development Lab, London, 2008.

[2] C.Ronchi, “The Practical Security Of Out-Of-Band Transaction Verification Methods”, e-Crime congress 2010, London, 2010.

[3] http://www.trusteer.com/product/trusteer-rapport (Access Date: 10 February 2011)

[4] T. Weigold, et al, “The Zurich Trusted Information Channel: An Efficient Defence against MITM and MS Attacks”, TRUST 2008, pp. 75-91.

[5] C.Ronchi, “Web Browser Hardening for Secure Internet Transactions”, RSAConference Europe 2009, London, 2009, Session ID: AND-303
http://www.h-browser.com (Access Date: 10 February 2011)

[6] http://www.crealogix.com/fileadmin/Leistungsangebot/pdf/products/e-banking/factsheet_ebs_en.pdf (Access Date: 10 February 2011)

[7] R. Oppliger, R. Rytz, T. Holderegger, “Internet Banking: Client-Side Attacks and Protection Mechanisms”, IEEE Computer, vol.42, no. 6, June 2009, pp. 27-33.

[8] C. Herley, “So Long, and No Thanks for the Externalities:The Rational Rejection of Security Advice by Users”, Proc.New Security Paradigms Workshop, ACM, 2009, pp. 133-144.

[9] B.Kerbs, “La. firm sues Capital One after losing thousands in online bank fraud“, The Washington Post, http://voices.washingtonpost.com/securityfix/2009/12/jmtest.html, 2009. (Access Date: 10 February 2011)

[10] D. Johnson, “Shames-Yeakel v. Citizens Financial Bank: Failure to Expeditiously Implement State-of the Art Security Measures Can Create Liability for Negligence in Data Breach Cases”, Digital Media Lawyer Blog.