Customizing SSL Certificate Extensions to Reduce False-Positive Certificate Error/Warning Messages

Şafak TARAZAN*‡, Atila BOSTAN**

* Electric and Electronics Engineering Department, Faculty of Engineering, Atılım University, Kızılcaşar Mah. 06836, İncek-Gölbaşı/Ankara/Turkey

**Computer Engineering Department, Faculty of Engineering, Atılım University, Kızılcaşar Mah. 06836, İncek-Gölbaşı/Ankara/Turkey

‡Atılım University Kızılcaşar Mah. 06836, İncek-Gölbaşı/Ankara/Turkey, Tel: +90 312 686 8792, Fax: +90 312 586 8091, e-mail: tarazansafak@gmail.com

Abstract- In today's Internet world, X.509 certificates are commonly used in SSL protocol to provide security for web-based services by server/client authentication and secure communication. Although SSL protocol presents a technical basis, this web-security largely depends on user awareness of security measures as well. There are significant number of scientific studies in the literature reporting that the count of invalid or self-signed certificate usage in today's Internet can not be overlooked. At the same time, quite a number of studies place emphasis on the acquired indifference towards certificate warning messages which are popped up by web browsers when visiting web pages with invalid or self-signed certificates. In this study, with the importance of user's daily practices in developing habits in mind, we studied a modification of X.509 certificates in order to reduce the number of false-positive certificate-warning pop ups in order to reduce gaining faulty usage habit of invalid certificates.

Keywords- X509 certificates; SSL protocol; certificate extensions; invalid certificates; SSL certificates and users awareness.

1. Introduction

The internet is a system which allows exchange of data among networked computing devices and it has an unquestionable role in daily life such as: education, business, gaming, military and so on. Most of the companies, regardless of the size, handle their process management and services on Internet. Even money transactions are being handled online between companies and organizations. Since most of the data exchange operations is done on public networks, a security mechanism is required in order to provide communication security and mutual trust between linked devices and systems.

By the development of the Internet technology most of the offline systems have been replaced with online systems whose data exchange structure commonly relays on web services and machine to machine communications. In this structure, in order to provide communication confidentiality and client/server authentication, SSL (secure socket layer) protocol[$] [1] is widely used. By its design, SSL protocol makes use of PKI (public key infrastructure) technique which establishes cryptographic-key exchange and a trust mechanism between two parties. SSL protocol is proved to be vulnerable to a variety of security

[$] Although TLS is an IETF standards track protocol (last updated in RFC 5246), that was based on the earlier SSL specifications developed by Netscape Corporation, throughout this article SSL is used to refer the technique used in both TLS and SSL.

attacks [2, 3, 4]. This technique is not fully autonomous and user awareness still has a remarkable role on controlling and accepting the invalid certificates. In other words, the user may break trust mechanism between parties (intentionally or unintentionally) and this provides grounds for an attacker to reach sensitive data. In this mode of certificate control, user awareness has a key role. Some scientific studies on the importance of user awareness in web-security are referred in following paragraph.

Most of the time, security precautions and tools are seen to be time-consuming and unnecessary in information systems which implicitly lead employees to overlook to security operations. Studies show that, although the importance of the security is mentioned by both academicians and government authorities, not enough attention is paid. There are also some studies supporting that end users weaken the security with following line. In the Computer Security Institute's reports [5, 6, 7, 8] web spoofing was reported among the most observed violations. Lack of security awareness in user behaviours is generally attributed as the rationale for most of security breaks [9]. The root causes of this user unawareness are stated with the following sentences. However, user awareness training was found to be the least significant one (given the allocated resources) in real security applications [10]. Even if the users are technically aware of the measures that should be followed to ensure secure usage, they are still subject to misuse. There are researches pointing out that technical awareness is not sufficient in providing secure usage, and end-user behaviours are not always consistent with their beliefs [11, 12, 13]. It is obvious that security awareness should be increased to eliminate behavioural security risks [13]. However, this is not an easy task due to the limited technical knowledge of the users. In order to reduce above mentioned user impact on the SSL security, there researches currently going on [14, 15],

In this study, we focused on removing false-positive certificate warning messages since warning messages are not very effective for users due to their limited knowledge and interest towards warning messages. In other words, we proposed a solution to reduce negative effect of the warning messages displayed to user due to invalid certificates. With our proposed solution, user dependency in SSL certificate verification mechanism is broken. However, our aim was not to eliminate black hat hackers taking advantage of MitM attack. For this purpose, we suggest a modified browser action on invalid and self-signed certificates by making use of certificate-extensions field in X.509 digital certificate structure. Proposed modification and certificate-extension usage is backward compatible with current web browsers as well. This study and reported implementation details are presented as a proof of concept. In other words, this study demonstrates the feasibility and applicability of the proposed usage in SSL web services.

An outline of background and related studies in the literature are given in Section 2 and 3, respectively. Technical and implementation details for the proposed structure are explained in Section 4. Finally, in Section 5, advantages, drawbacks and constraints in proposed-usage are reported in the conclusions.

2. Background and Related Work

In SSL protocol, X.509 certificates stand for data structures, given in Figure 1, which binds required public key to the related subject for the key exchange mechanism. The binding is provided by having a trusted CA digitally sign each certificate [16]. These certificates are used at the beginning of the handshake process for peer authentication and communication encryption.

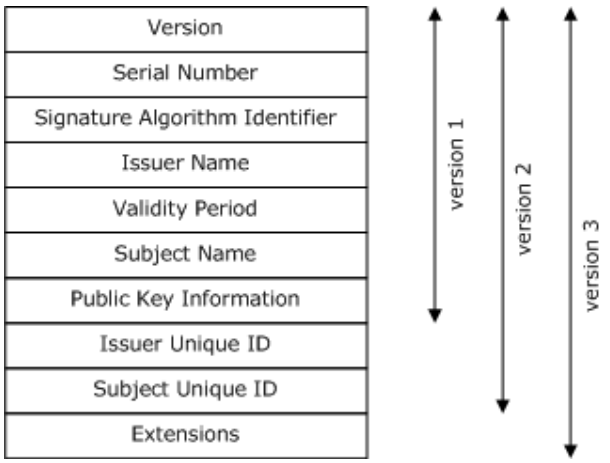

Fig. 1. X509 Certificate Structure

In the SSL handshake process, the client receives an SSL certificate from the server. After this operation, the client counts on its trust anchors to verify the certificate [17]. Depending on the validity of the certificate, the handshake proceeds or terminates. Although this X.509 certificates and SSL protocol aim to prevent security breaches, there are still risks to be concerned. Following lines summarize academic researchers related to these risks.

As it was indicated in security reports and surveys, phishing and web-spoofing are still among the most observed security breaches [5, 6, 7]. To prevent this kind of security issues and provide a better level of security, (SSL) is developed. Secure Sockets Layer (SSL) is a protocol used for verifying identification of a website and securing the communication between peers [17]. Although SSL is provided as an automated mechanism, it is not fully automated in case of invalid certificates. In such a case, the user is expected to decide on whether to trust an invalid certificate or not by checking a technical warning message displayed. Most of the studies agree on that users take no precautions against information security attacks, even basic ones. Similarly, researchers emphasize that users are oblivious to security cues, ignore certificate error/warnings and cannot tell legitimate web-sites from phishing imitations [18].

Besides user behaviours on security precautions, warning/error messages and their effectiveness is also studied to provide more reliable communication. In a phishing experiment conducted by Dhamija et al. [19] it is stated that most of the users were tricked by phishing due to the incompetent of the security icons and insufficient knowledge of the users about the indicators [20]. Additionally, Microsoft researchers spotting on using different icons and texts on warning messages considerably effect on users' risk perspicacity [21]. On the other hand, as Microsoft researcher Mr. Harley summarized, we can't complain about users' limited attention to SSL certificates due to false-positive warning/error messages caused by invalid certificates [22]. However, it can't be ignored that SSL warning/error messages are considered to be as a part of the website by users because they see them at even reliable websites [23]. As, warnings science literature suggests, warnings and error messages should be used as last option unless we have an approach to reduce possible threat and hazard [24].

3. Proposed Solution

When we consider X509 Certificate's format, it has optional "extensions" of which the purpose is given in the RFC 3280 section 4.2 as follows: "The extensions defined for X.509 v3 certificates provide methods for associating additional attributes with users or public keys and for managing a certification hierarchy". Each extension consists of an OID (Object Identifier), Criticality and ASN.1 (Abstract Syntax Notation One) structured string value. OID represents the id of the extension which has ASN.1 structured string value and Criticality states how should be the systems response if the extension is not recognized by having value critical or non-critical. If Criticality field set as critical and system does not recognize the extension, certificate must be rejected. However, if a system encounters a non-critical extension and doesn't recognize it, still certificate can be accepted. These extensions can be categorized in too groups; standard and custom extensions. The term standard extension states the X.509 v3 defines a widely applicable extension to the X.509 v2. On the other hand, there are numerous reasons why customization of the extension data is required in some cases, for this reason, in X.509 v3, customized data can be inserted in the certificate with a registered OID which is called custom extension.

In normal scenario while validating X.509 certificates, following steps are checked at client side; certificate signature-certificate is issued by a certificate authority or not-, certificate expiration date, certificate revocation status and certificate-browser URL matching. If certificate passes all these verification steps, handshake process begins and communication starts between peers. However, if the given certificate's validation fails in any of steps mentioned above, a warning message with an option is displayed to the user to

confirm the action to be performed; terminate the connection or continue.

As stated in the previous section, displaying such a warning message to user implicitly enables him/her to break certificate verification. Since users are unable to understand the risks stated by the error messages, so that they can not take the required actions. And yet, this is not rare case. Since most organizations prefer SSL web services only for communication confidentiality, but not for peer authentication, this mode of operation brings about security breaches in terms of user experience with error messages due to self-signed/invalid certificates.

We suggest freeing from this conflicting status by adding a custom extension to certificate which enables browsers or client applications to distinguish desired usage purpose of the certificates such as; only for encryption of the communication or both encryption of the communication and peer-authentication. In other words, adding a custom extension to the certificate with unique OID and ASN.1 structured value brings an opportunity to use certificate only to encrypt the communication or provide both encryption of the communication and authentication of the peers. By this way, certificates, intended to provide encryption only, can be interpreted as valid, with an indicator in the browser stating that certificate is encryption only, on the client side and client's application would not display any warning message to user, since in this case certificate does not need to be issued by a trusted certificate authority. Moreover, if a client application encounters with an invalid certificate on the https protocol, in which the certificates are used both for communication-encryption and peer authentication, user can be redirected to available http protocol by displaying nothing but only an informative message to user. Unlikely, if http service is not online on the same server, communication should be broken by client application with a communication error message shown to the user. With this mode of operation, in any way, user would not have an option to keep the communication while something is wrong in an https connection. In addition to this, since the extension's criticality flag was set to be non-critical, if a client application doesn't understand the inserted flag, client system should ignore the flag and continue on its old procedure, as it is in the current browser operations. These properties of certificate extensions fill will backward capability with the old client systems and browsers.

4. Implementation Details

As a proof of concept there are two parts to be considered; generating X.509 certificate with custom extension and handling this certificate on the client side. In this implementation, certificate was generated using OpenSSL 1.0.1j (64 bit). Certificate handling and verification was performed using Visual Studio .Net Windows Forms Application.

As the first part of the study, a X.509 certificate was created with custom extension in it. To add custom extension to certificate using OpenSSL, openssl.cfg file modified with desired extension by writing them below to [v3_req] tag. In this experimental study, the code below was added where "1.2.3.412" stands for OID of the extension which should be registered by a formal request from authorities such as:IANA (Internet Assigned Numbers Authority) or ISO Name Registration Authority. By default, extension criticality is set to "non-critical". When critical extension is needed, the reserved word "critical" must be added to extension. In the given lines of code, "BOOLEAN" represents variable type and "ASN1" represents variable's notation is in ASN1.

```
1.2.3.412=ASN1:BOOLEAN:TRUE
//for default non-critical extension declaration

1.2.3.412=critical,ASN1:BOOLEAN:TRUE
//for critical extension declaration
```

According to the openssl.cfg file, sample certificate was created with desired non-critical custom extensions. The command line command to create certificate by Open-SSL is given below:

```
openssl.exe  x509 -req -days 365 -in server.csr -signkey server.key -out server.crt -extensions v3_req -extfile openssl.cfg
```

On the second part of the implementation, a Https server was set up on a Windows 7 64 bit computer. In this server, generated X.509 certificate, which contains custom extension, was used in SSL protocol. On the client application,

while validating the certificate, inserted custom extension OID was searched and checked. If it was set to be encryption only, browser brought the content without showing any error/warning message to the user. On the other hand, if browser encounters an invalid certificate where specified OID was set to be both encryption and authentication, current https connection was broken and user was redirected to available http version of the application with displayed an informative message to user. Flow diagram of suggested control is given in the Figure 2.

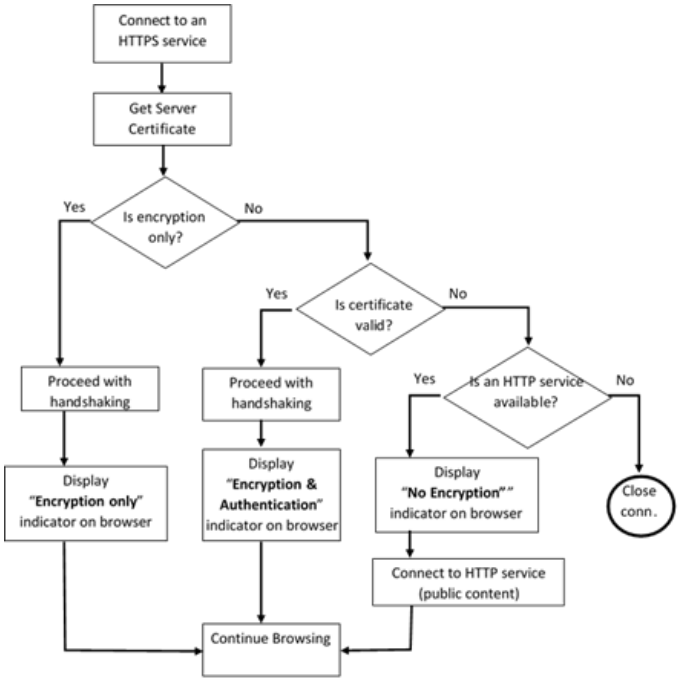

Fig. 2. Suggested Control Flow Diagram

4.1. Practical Usage and Typical Scenarios in Action

The security requirements in publishing a web site can be classified in tree types. There may be no security-sensitive content in the pages, so that it is safe to publish the site on an HTTP service. On the other hand, the web site may contain some security-sensitive data which needs the publisher (service provider) be authenticated and communication between this site and any visiting client be encrypted in order to guarantee the service is authentic and prevent any eavesdropping. In this case, an HTTPS service, by the book, should better be used. But in some cases the owners or administrators of the web site prefer encrypting the traffic between server and visiting clients only, while sacrificing the server authentication. A realistic example may be a web page which transmits log-in information (user name and password) from client to server but the content of the web pages are all unclassified. We concluded this last type of security requirement out of significant number of reported invalid certificate usage in SSL web services on the Internet [4, 5, 6, 7, 8]. This number of invalid certificate usage should indicate a common requirement in web publishing. Since, there is considerable number of HTTPS services running with self-signed certificates. For this third type of security requirement there is no direct technical solution yet. So that, the site administrators, in this third type of a security-requirement, opt for using self-signed certificates in their SSL web services while facing the clients with false-positive certificate error messages.

The technique proposed in this paper suggests a solution for above mentioned third type of security-requirement with an intention to cease the false-positive certificate error messages. Although, in this paper we propose a solution to indicate the certificate usage purpose of web site administrators only, it would be explanatory to elucidate how this technique will be helpful in preventing the false-positive certificate error messages with some generic scenarios.

Scenario 1: Web site contains classified information, the server needs to be authenticated and the communication should be encrypted.

In this scenario, site should better be published on an HTTPS service with a valid digital certificate (issued by a trusted authority) which is authorized for authentication and encryption (both). Browsers are supposed to display an appropriate marker (a colour in the address or status bar / a padlock icon etc.) when visiting the site. If an malicious user wants to set a MitM attack by using a copy of the original site certificate then browser will end the connection during the SSL handshake because of the mismatch between URL and the name in the certificate.

Scenario 2: Web site contains classified information and the server does not need to be

authenticated but the communication should be encrypted.

In this scenario, site can be published on an HTTPS service with a digital certificate created by a trusted authority or self-signed, but the certificate is to be authorized only for encryption. However, the browsers should display an appropriate marker (may be a pop-up window / a colour in the address or status bar / an overlay icon etc., or any combination of these) which informs the user "Server site is not authenticated but the communication is encrypted). In this scenario, if a malicious user wants to set a MitM attack the security risk is not more than the current implementations, since in the current usage it is in user's discretion to continue or abort the connection. Invariably, user is to decide to continue or abort the connection in the proposed usage. The difference between the current and the proposed implementations is in the warning message. In current usage a certificate error message is displayed to the user, and most of the time a regular user generally does not understand what is wrong with site-certificate and does not have enough resources to verify if there is an attacker in the middle. Furthermore, most of the users does not have any idea what are the security risks they are undertaking when they click "Continue" on certificate warning window. In the proposed usage, if the certificate in use is authorized only for encryption and in valid-period, issued to the active server name (URL match), passes signature verification (content not changed) then the browser is supposed to inform the user on that "the server is not authenticated but the communication is encrypted". It is again in user's discretion to continue or abort the connection. The advantage in the proposed usage is when the certificate fails in any one of the control steps other than trusted-issuer verification the connection will be closed without any user discretion. Undoubtedly, the most important contribution of the proposed usage will be cancellation of "something is wrong with the certificate. Do you want to continue?" type of a warning message was proven to induce indifference in the users.

In the proposed usage, directing the user to an HTTP service when the certificate controls fail is suggested in order to keep unclassified content publication available if the administrator of the site prefers (such as unclassified pages without any log-in dialogs). In practice it is very common to use HTTP services to direct the users to HTTPS service. Although it is beyond the scope of this study, here we present how this user directing issue may be resolved. One of the alternative solution may be including a default procedure in browsers which tries to connect the same URL but by using an HTTP connection when a certificate error is detected on an HTTPS service. The other solution may be defining a special HTML metadata element which points to the unclassified HTTP service in case certificate is found to be faulty. In either solutions taking the advantage of specialized cookies will help to identify the response direction jumps.

5. Conclusion

Indisputably, Internet plays a significant role in all daily activities such as; education, politics, health and trade where security is one of the biggest concern to protect sensitive data or privacy. In order to increase security, X.509 certificate based PKI was introduced and being used to provide communication encryption and client/server authentication in SSL-web services. However this technique is not fully automated and in some cases user awareness and actions still have important role. When a system encounters an invalid certificate, displays a warning message and asks user to continue or terminate by breaking the autonomous control chain of certificate verification. This is very common situation in case of widely used self-signed SSL certificates since CA signed certificates are expensive. With the high number of non-conforming certificates in SSL-web services on the Internet, users develop indifference towards certificate warning messages and assume keeping the connection on brings in no security risk at all.

In this study, we have studied elimination of error messages due to invalid SSL certificates by modifying the X.509 certificate. Basically, we added custom extension to bring better automation to certificate control operation. With this proposed mechanism, warning message no longer will be displayed to user. Moreover, if a system

encounters with an invalid certificate, https connection will be broken and user will be redirected to http protocol of the service if it is available. By this way, automation in certificate based SSL service will be increased. Moreover, option of keeping an https connection while the certificate is invalid will never be available to the user; hence possible security risks which would stem from lack of user awareness of carelessness will be avoided. Although we propose a certificate extension and revised browser behaviour, with the intention to reduce user interaction in establishing the security in SSL web communication, this mode of a configuration will still have old security risks as well. For instance, a man-in-the-middle attack with a self-signed certificate could be intruded in a server-side authenticated SSL communication by establishing an authenticated connection with server-side while having a self-signed (encryption-only) connection with the client-host. With this mode of usage, attacker is able to perform man in the middle attack. However, in this study, our focus was gained user habits with the SSL error messages; not to eliminate MitM. With parallel to this study, we are now also performing researches on if any technical solution which could prevent or detect this attack in our proposition is possible or not.

After this technical proof of concept and prescience, as a future work, we are planning to focus on removing the mentioned MitM attack in the suggested configuration and proving influence of this approach on users where we can analyse and compare certificate usage habits in both current implementation and proposed implementation in this paper.

References

[1] T. Dierks, The transport layer security (TLS) protocol version 1.2, IETF RFC-5246, 2008, Available online at https://tools.ietf.org/html/rfc5246.

[2] K. Paterson and M. Albrecht, "Lucky Microseconds: A Timing Attack on Amazon's s2n Implementation of TLS", Real World Cryptography Conference 2016, 6-8 January 2016, Stanford, CA, USA.

[3] V. K. Keerthi, "Taxonomy of SSL/TLS Attacks.", International Journal of Computer Network and Information Security, Vol.8 No 2, Feb. 2016

[4] X. D. C. de Carnavalet and Mannan, M., "Killed by Proxy: Analyzing Client-end TLS Interception Software.", Cocordia university publications, 2016, http://users.encs.concordia.ca/~mmannan/publications/ssl-interception-ndss2016.pdf, Latest Access Time for the website is 23 April 2016.

[5] V. S Subrahmanian, M. Ovelgonne, T. Dumitras and A. Prakash, The Global Cyber-Vulnerability Report., ISBN: 978-3-319-25758-7, 2016.

[6] CSI 2010-2011, 15th Annual CSI Computer Crime & Security Survey, Computer Security Institute, 2011, http://reports.informationweek.com/cart/index/downloadlink/id/7377, Latest Access Time for the website is 12 December 2013.

[7] CSI 2009, 14th Annual CSI Computer Crime & Security Survey, Comprehensive Addition, Computer Security Institute, 2009, http://gocsi.com/purchase_survey, Latest Access Time for the website is 11 June 2011.

[8] CSI 2008, CSI Computer Crime & Security Survey (2008), Computer Security Institute, http://gocsi.com/sites/default/files/uploads/CSIsurvey2008.pdf, Latest Access Time for the website is 12 December 2013.

[9] P. Kamal, "State of the Art Survey on Session Hijacking.", Global Journal of Computer Science and Technology, Vol.15, No.1, 2016

[10] J. D'Arcy and A.Hovav, "Deterring Internal Information Misuse", Communications of the ACM, Vol.50 No.10, pp 113-117, October 2007

[11] Kevin Palfreyman and Tom Rodden, "A Protocol for User Awareness And World Wide Web", Proceedings of Computer Supported Cooperative Work'96, Cambridge MA, USA,1996, ACM 0-89791-765-0/96/11

[12] B. Gross Joshua and B. Rosson Mary, "Looking for Trouble: Understanding End-User Security Management", Computer Human Interaction for Management of IT (CHIMIT'07), Cambridge MA. USA., 30-31 March 2007, ACM 1-59593-635-6/97/0003

[13] M. Evans, L. A. Maglaras, Y. He and H. Janicke, "Human Behaviour as an aspect of Cyber Security Assurance.", arXiv preprint arXiv:1601.03921, 2016

[14] Hugo Krawczyk and Hoeteck Wee, "The OPTLS Protocol and TLS 1.3", Real World Cryptography Conference 2016, 6-8 January 2016, Stanford, CA, USA.

[15] Adrienne Porter Felt, "Where the Wild Warnings Are: The TLS Story", Real World Cryptography Conference 2016, 6-8 January 2016, Stanford, CA, USA.

[16] Shuhaili Talib, L. Clarke Nathan and M. Steven Furnell, "An analysis of information security awareness

within home and work environments.", Availability, Reliability, and Security (ARES'10), International Conference on. IEEE, 2010.

[17] Henry Story, B. Harbulot, I. Jacobi and M. Jones, "Foaf+ ssl: Restful authentication for the social web.", Proceedings of the First Workshop on Trust and Privacy on the Social and Semantic Web (SPOT2009). June 2009.

[18] Jennifer Sobey, P. C. Van Oorschot, and Andrew S. Patrick, "Browser Interfaces and EV-SSL Certificates: Confusion, Inconsistencies and HCI Challenges.", Carleton University School of Computer Science, Canada, Technical Report TR-09-02, 15 January 2009.

[19] Devdatta Akhawe and Porter Felt Adrienne, "Alice in Warningland: A Large-Scale Field Study of Browser Security Warning Effectiveness.", Usenix Security. 2013, Washington DC. USA, 14-16 Augustos 2013, pp 257-272

[20] R. Dhamija, J. Tygar and M. Hearst, "Why Phishing Works", Proceedings of the Conference on Human Factors in Computing Systems (CHI), New York, NY, USA, p. 581- 590, 2006.

[21] T. S. Amer and J. B. Maris, "Signal words and signal icons in application control and information technology exception messages – hazard matching and habituation effects.", Tech. Rep. Working Paper Series–06-05, Northern Arizona University, Flagstaff AZ. USA, October 2006.

[22] Herley Cormac, "So long, and no thanks for the externalities: the rational rejection of security advice by users.", Proceedings of the Workshop on New Security Paradigms, ACM 2009, Queen's College, Oxford, UK.

[23] Serge Egelman, Trust me: Design patterns for constructing trustworthy trust indicators.", ProQuest, 2009.

[24] J. Sunshine, S. Egelman, H. Almuhimedi, N. Atri and L. F. Cranor, "Crying Wolf: An Empirical Study of SSL Warning Effectiveness.", 18th USENIX Security Symposium, San Jose CA. USA, pp 399-416, 10-14 August 2009.