Chapter 24

# EXTENDED WEB SITE SECURITY FUNCTIONALITY

## INTRODUCTION

The web site is the medium for an increasing amount of business and other sensitive transactions, for example for online banking and brokerage. Virtually all browsers and servers deploy the SSL/TLS protocols to address concerns about security. However, the current usage of SSL/TLS by browsers, still allows extended web site functionality spoofing (misleading users by impersonation or misrepresentation of identity or of credentials) [1].

Indeed, there is an alarming increase in the amount of real-life web site-spoofing attacks, usually using simple techniques. Often, the swindlers lure the user to the spoofed web site (impersonating as financial institution, by sending her or him spoofed e-mail messages that link into the spoofed web-sites); this is often called a phishing attack. The goal of the attackers is often to obtain user-ID's, passwords/PINs and other personal and financial information, and abuse it (for identity theft). A recent study by industry analysts found that about six million users gave such information to spoofed web sites, and estimate 5.6B$ direct losses to U.S. banks and credit card issuers during 2006; other estimates of yearly damage due to phishing and spoofing attacks are between $900 million to $1.4 billion. Spoofing attacks, mostly using the phishing technique, are significant threats to secure e-commerce [1].

This chapter investigates spoofing and phishing attacks and present countermeasures, with regards to extended web site security functionality, while focusing on solutions that protect naïve as well as expert users. These attacks are not easy to deploy, as they require technical sophistication, and are sensitive to changes in browsers, operating systems and their configurations [1].

In fact, practical web site-spoofing attacks deployed so far, do not use such techniques, or use just basic scripts and browser vulnerabilities (to present fake location bar. Almost all of the many reported attacks left significant clues for the expert, attentive user, such as the lack of use of SSL/TLS (indicated by an open padlock icon, or by the lack of a padlock icon),

and/or the use of a URL from a domain not owned by the victim web site. Such attacks will therefore succeed, even when using countermeasures [1].

Still, these simple attacks are very effective. It is argued that this is due to several weaknesses in the user interface of the current popular browsers, and suggests simple extensions to browsers, that should prevent many or most of these attacks. The goal here is to improve browser security-related user interface (UI), and protect (even) naïve, inattentive web users [1].

The first principle establishes the importance of default settings related to security, such as the list of certification authorities trusted by browsers. This is a special case of the unmotivated user principle and of the path of least resistance principle [1].

## SECURE UI PRINCIPLE I: SECURITY SHOULD BE DEFAULT, AND DEFAULTS SHOULD BE SECURE

Default settings should provide adequate security, and only globally-trusted, obviously trustworthy parties may be trusted by default. Even if defaults are secure, users may overrule them and disable security features, if they are overly disruptive to normal work and annoying, and in particular if their importance is not sufficiently clear to the users, and especially if disabling is easy. For example, many browsers, by default, warn users if an unprotected web page contains a form (whose contents will be sent in the clear). However, most web-forms, currently, are not protected; therefore this message pops up very frequently and almost all users disable it (do not display this warning any more). This leads to the next principle [1]:

## SECURE UI PRINCIPLE II: SECURITY MUST BE USABLE TO BE USED

Users will avoid or disable security mechanisms which are hard to use, disruptive or annoying. Secure usage should be the most natural and easy usage [1].

The next secure UI principle follows from well-known user-interface design principles such as recognition rather than recall principle. Its relevance to security was noted by observing that users tend to click thru textual warning messages (the unprotected form warning previously mentioned). Also, important sites such PayPal, Microsoft's Passport, Yahoo!, e-Bay and Chase, all ask users to enter passwords and/or other sensitive information in insecure web pages; this shows that not only users, but also web site designers and auditors did not notice the lack of protection [1].

## SECURE UI PRINCIPLE III: ALERTS SHOULD WAKE-UP

Indicate security alerts (potential security exposures), using a clear, interruptive audio/visual signal. Alerts should not only be noticeable and interruptive, they (and other