Trusted User-Aware Web Authentication

Sebastian Gajek, Ahmad-Reza Sadeghi, Jörg Schwenk,
and Marcel Winandy
Horst Görtz Institute for IT Security
Ruhr-University Bochum
Universitätsstr. 150, D-44780 Bochum, Germany

Password theft and identity fraud are a challenging problem to deal with when using Internet services. Various reasons have been identified for this, e.g., users do not understand security indicators of their web browser application and cannot distinguish a legitimate from a faked web site. In any case, web authentication is a ceremony [7] that centers around the user and her skills: The user decides finally whether connections to the Internet are trustworthy. As recent studies point out [6, 14], we cannot rely on this trust model anymore.

Researchers have presented many innovative and promising mechanisms to counter identity fraud and to lower trust assumptions. Examples are enhanced authentication protocols [13, 10] as well as user interfaces that are spoof-resilient [1, 5] and indicate security in a way that average Internet users without background in cryptography can understand [3, 8]. Unfortunately, the user is still involved in each ceremony of authentication, and as human users are prone to errors, a number of users can still be tricked by an attacker despite the protection mechanisms. Hence, we wish the authentication to become less dependent on the user and her skills, and carried out to an authentication agent which is less susceptible to failures. Ideally, once such an agent is set up, it shall authenticate the user and relieve her from any burden (e.g., verify security indicators, type password). Some research has already been done to automate web authentication; example instantiations are simple password managers (e.g. KDE's KWallet) or more sophisticated identity providers like Web Wallet [15] and CardSpace [12].

However, as the technical sophistication of attacks increases, the trust assumptions made on the underlying operating environment also change: Several researchers expect a growth in different types of malware [11, 9, 2]. Specifically designed Trojan horse programs are deployed by attackers, e.g., keyloggers, which destroy any confidence a user may have when she interacts with a password interface. Thus, to provide secure web authentication in the future, we cannot only rely on new authentication protocols or browser enhancements, but we must also take the operating system environment into account. We need to preserve the confidentiality and integrity of identity-critical data and applications. Moreover, a trusted path must be established to those interfaces where we enter our credentials. Unfortunately, commodity operating systems do not fulfill these requirement due to lack of security functionality on the one hand and due to their complexity on the other hand. But the compatibility to legacy systems is an important requirement for any new system to be widely deployed.

Virtualization provides an efficient means for isolating critical applications from others while allowing the interoperability and re-use of existing operating systems and applications. Approaches like Tahoma [4] showed how to use virtualization in order to isolate browser instances from each other, or how to isolate an authentication agent from the browser as realized in SpyBlock [9]. We propose a modular security architecture and reference implementation which integrates and enhances approaches based on identity providers (like password managers) but also provides protection against malware and against interface spoofing like picture-in-picture attacks. Our approach is based on the idea of compartmentalization for isolating applications of different trust level, i.e., we separate the web usage in the browser from the web authentication. For the latter, we use a trusted wallet in a separated compartment to store the user's credentials and to authenticate sensitive services as a proxy on behalf of the user. It does not require specific skills from users to distinguish between real and faked web sites by identifying security indicators. The user has only to store credentials in the trusted wallet once. We discuss how to setup and update credentials and how to solve problems that may arise when security-unaware users want to apply the same credentials to different services.

In contrast to existing proposals our solution provides protection of the whole operating environment, not only the applications within the virtual machines. To confirm the security guarantees of integrity and confidentiality, we use a security kernel that is executed on hardware that supports Trusted Computing functionality as provided by, e.g., a TPM[1], as today several computer manufacturers already ship their computer platforms equipped with a TPM. This allows us to establish a secure bootstrap process of the trusted computing base and to bind identity-related sensitive data to the integrity of the wallet application and its execution environment.

Moreover, we revisit the concepts of secure graphical user interfaces. We show that due to virtualization the security kernel can easily provide a secure user interface that provides confidentiality (e.g., against keyloggers) and enables the user to authenticate compartments and clearly distinguish between trusted and untrusted compartments. We therefore use a reserved real estate to authenticate the compartment the user currently interacts with. This reserved area cannot be modified by any compartment.

Besides the sketched protection mechanisms and the compatibility to legacy systems, our approach has the advantage that the web browser application does not need any modification or add-on and that the service providers do not need to change their systems or protocols.

[1] Trusted Platform Module as specified by the Trusted Computing Group.

References

[1] A. Adelsbach, S. Gajek, and J. Schwenk. Visual Spoofing of SSL Protected Web Sites and Effective Countermeasures. In *Information Security Practice and Experience Conference*, 2005.

[2] D. Birk, S. Gajek, F. Grobert, and A.-R. Sadeghi. Phishing phishers—oberserving and tracing organized cybercrime. In *Proceedings of the IEEE CYBER-FRAUD Workshop*, 2007. (to appear).

[3] N. Chou, R. Ledesma, Y. Teraguchi, and J. C. Mitchell. Client-Side Defense Against Web-Based Identity Theft. In *NDSS*, 2004.

[4] R. S. Cox, S. D. Gribble, H. M. Levy, and J. G. Hansen. A Safety-Oriented Platform for Web Applications. In *Proceedings of the 2006 IEEE Symposium on Security and Privacy (S&P'06)*, pages 350–364. IEEE Computer Society, 2006.

[5] R. Dhamija and J. D. Tygar. The Battle Against Phishing: Dynamic Security Skins. In *SOUPS '05: Proceedings of the 2005 Symposium on Usable Privacy and Security*, pages 77–88, New York, NY, USA, 2005. ACM Press.

[6] R. Dhamija, J. D. Tygar, and M. A. Hearst. Why phishing works. In R. E. Grinter, T. Rodden, P. M. Aoki, E. Cutrell, R. Jeffries, and G. M. Olson, editors, *CHI*, pages 581–590. ACM, 2006.

[7] C. Ellison. Ceremonies. Crypto Rump Session, 2005.

[8] A. Herzberg. Protecting web users from phishing, spoofing and malware. Cryptology ePrint Archive, Report 2006/083, 2006. `http://eprint.iacr.org/`.

[9] C. Jackson, D. Boneh, and J. C. Mitchell. Attack of the transaction generators, 2007. (Manuscript).

[10] M. Jakobsson, S. Myers, and M. Augiere. Delayed Password Disclosure, 2005. `http://www.informatics.indiana.edu/markus/stealth-attacks.htm`.

[11] E. Levy. Criminals Become Tech Savvy. *IEEE Security and Privacy*, 02(2):65–68, 2004.

[12] R. Oppliger, S. Gajek, and R. Hauser. Security of mircrosoft's identity metasystem and cardspace. In *Proceedings of the 15th GI/ITG Conference on "Kommunikation in Verteilten Systemen" (KiVS '07)*, pages 63–74. VDE Verlag, Berlin, 2007.

[13] M. Steiner, P. Buhler, T. Eirich, and M. Waidner. Secure password-based cipher suite for TLS. 4(2):134–157, May 2001.

[14] A. O. Stuart Schechter, Rachna Dhamija and I. Fischer. The emperor's new security indicators, 2007. to appear in the Proceedings of the IEEE Symposium on Security and Privacy.

[15] M. Wu, R. C. Miller, and G. Little. Web Wallet: Preventing Phishing Attacks by Revealing User Intentions. In *SOUPS '06: Proceedings of the Second Symposium on Usable Privacy and Security*, pages 102–113. ACM Press, 2006.