Passwords for Both Mobile and Desktop Computers

Appendix (online only)

MOHAMMAD MANNAN AND P.C. VAN OORSCHOT

Mohammad Mannan is an Assistant Professor at the Concordia Institute for Information Systems Engineering, Concordia University, Montreal. He has a PhD in Computer Science from Carleton University (2009) in the area of Internet authentication and usable security. He was a postdoctoral fellow in the Department of Electrical and Computer Engineering at the University of Toronto from 2009 to 2011 (funded by an NSERC PDF, and ISSNet). His research interests lie in the area of Internet and systems security, with a focus on solving high-impact security and privacy problems of today's Internet.
mmannan@ciise.concordia.ca

P.C. van Oorschot is a Professor of computer science at Carleton University in Ottawa, where he is a Canada Research Chair in Authentication and Computer Security. He was program chair of USENIX Security 2008, program co-chair of NDSS 2001 and 2002, and co-author of the *Handbook of Applied Cryptography* (1996). He is on the editorial board of IEEE TIFS and IEEE TDSC. His current research interests include authentication and identity management, security and usability, software security, and computer security. He is a member of the IEEE.
paulv@scs.carleton.ca

Other Proposals Supporting Password Entry on Mobile Devices

The Blue Moon authentication scheme [5] is a preference-based secondary login system designed for when users forget account passwords. Its primary goal is to replace common password reset methods, such as personal verification questions (PVQs), by using personal preference-based questions; the underlying assumption is that preferences are more stable than long-term memory. During setup, users select items they like and dislike from several categories (e.g., sports, music, food). For authentication, users must correctly categorize previously selected items as "liked" or "disliked." Items can be presented as text or image; an image-based implementation is available for mobile devices (http://mobile-blue-moon-authentication.com/). For both the text-based and image-based schemes, users need only select displayed items, e.g., via mouse pointer or touch, rather than by typing. Thus Blue Moon seems appropriate for primary Web logins in both desktop and mobile platforms, if adopted by site maintainers.

To ease text entry, many smartphone platforms offer a predictive text entry feature where the system auto-fills or suggests a list of commonly used words once a user has typed the first few characters. At times, auto-correction may produce amusing results; see, e.g., http://damnyouautocorrect.com/. Cheswick [3] has recently revived the circa 1980s or earlier idea of multi-word passwords to combine this and users' existing preference to choose dictionary words as passwords; see Bicakci and van Oorschot [1] for a summary of old and new variations of multi-word password proposals. The basic idea of multi-word passwords [3] is that instead of a non-dictionary password with special characters, users (must) choose multiple common words as their password. Users need type only a few characters per word of their multi-word password, enabling easy-to-input, high-entropy passwords for mobile devices. For example, a system-assignment of three words from a fixed 1024-word list provides 30 bits of entropy; the password distribution, and thus entropy, should be expected to be skewed if selection from the list is user-chosen. Predictive text is also easily implemented on desktop platforms; desktop browsers now commonly integrate dictionaries to help users fill forms. Multi-word passwords thus appear to offer convenient password entry on both platforms—if adopted by Web sites—although we are aware of no user studies that explore their memorability or usability in general.

Password patterns on a 9-dot grid used for screen-unlocking of Android phones are a simplified form of graphical passwords, used for local device authentication. This authentication mechanism is not presently compatible with desktop machines, which typically (at present) lack touch-sensitive screens, although compatible mouse-driven interfaces could easily be implemented. The PIN-level security seems mainly of interest for casual security appropriate for screen-locking rather than remote authentication to Web sites.

Jakobsson et al. [4] developed a model for *implicit authentication* (IA), in which users are authenticated based on passively collected usage data from their mobile devices (e.g., phone calls, SMSes, GPS coordinates, email, and calendar events). During authentication, IA outputs a score comparing the user's recent usage behavior with a pre-established *user model* (calculated from the user's past usage data); this score is then used to make authentication decisions. Note that IA mechanisms are already being used in the desktop world by certain industry sectors; see, e.g., RSA adaptive authentication (http://www.rsa.com/node.aspx?id=3018).

To secure the increasing amount of sensitive data on mobile devices against malicious apps, the on-board credential (ObC [6]) system has been developed by Nokia for devices running on Symbian and Maemo systems. ObC secures user credentials by relying on trusted hardware such as Trusted Platform Module (TPM) and M-Shield.

Syncing Objects and the Case for Allowing Multiple Passwords

Password objects must be copied (or made available via other methods, including portable memory cards) to all devices/platforms from which the user wishes to use ObPwd; and synced if passwords are updated. Existing sharing and sync mechanisms can facilitate the availability of the same password-generating object on these platforms. For example, Digital Living Network Alliance (DLNA) certified devices—including TVs, DVD players, game consoles, computers, and mobile devices—can easily share digital content such as photos, video, and music files when connected in a home network. Millions of such devices are currently in use; see dlna.org. Advanced sync techniques for generic user content have also been proposed, e.g., *device transparency* [7]. Note that, to prevent exposure of password objects, which is equivalent to leaking real passwords, the sync mechanisms must be over secure channels (e.g., physical USB connection, encrypted connection over Bluetooth or WiFi). In the absence of such guarantees, this makes reliance on generic syncing tools dangerous if ObPwd is used.

Although sync mechanisms are readily available in modern devices, we expect syncing of password objects would remain a usability and/or deployability obstacle for many users. Here we briefly sketch an alternative that will require back-end support. Beyond the current practice of supporting one valid password per account, multiple passwords could be allowed for accessing each account. Services can make available new interfaces that allow entering alternate passwords (e.g., as part of a "change password" dialogue, a new option would "allow an alternate password, e.g., from a mobile phone"); by entering the original password (or any later registered ones), users can authorize the use of the alternate password (with a customary second-time password entry for confirmation). Then users can use the same or different password objects from their multiple devices without syncing the objects. To some extent, this is akin to services that allow users to register alternate email addresses in case the primary address becomes inaccessible. Allowing multiple passwords may also encourage the use of more device-specific password mechanisms. However, we reiterate that this proposal breaks the "drop-in" feature of current ObPwd, which we believe is a huge enabling factor. The use of alternative passwords (instead of syncing) also increases the cognitive load for users, and, for example, increases the chance of errors due to password interference, i.e., confusing passwords between accounts.

Ratings Used in the Usability-Deployability-Security Evaluation

For the "Web passwords, mobile" row of Table 1 in the main article, we note the following ratings as downgrades from the desktop version: *Not-Efficient-to-Use*/U6 and *Not-Infrequent-Errors*/U7 (password entry is less efficient and more error-prone on mobile keyboards); and *Quasi-Accessible*/D1 (e.g., motor-impaired and blind users may have additional challenges). The rating *Nothing-to-Carry*/U3 is unchanged (the mobile platform is the primary device, not an auxiliary device needed for login to a primary device).

We rate "ObPwd, desktop" as *Quasi-Memorywise-Effortless*/U1 (users must remember where to find the password object file in their file system but need not remember precise syntax details of the password characters) and as *Scalable-for-Users*/U2 (one password object generates unique passwords for different Web sites). We rate it *Not-Nothing-to-Carry*/U3 in order to highlight the following device dependency: a desktop user may need to carry storage media containing their object files, for the reason of not having access to their digital objects on all login devices that they may wish to use (e.g., consider a friend's machine); available syncing mechanisms (as discussed earlier) should not be used on borrowed machines. We rate it *Not-Physically-Effortless*/U4, by this benefit's strict definition, as login requires more than the press of a button. It is *Easy-to-Learn*/U5. It is *Quasi-Efficient-to-Use*/U6, as locating a single object file becomes easier with repeated use (similar to typing the same password). Note that, in our user study [2], the ObPwd login task on average took almost twice as long as text password login; however, users were selecting different object files for their eight test accounts. We rate it *Infrequent-Errors*/U7 (typing errors are eliminated; users need to locate only one password object across accounts). We rate it *Easy-Recovery-from-Loss*/U8 (if a user forgets the object file's path or loses the object file, recovery is possible by the same mechanism as for lost regular text passwords).

For the primary use case of image-based objects, we must rate ObPwd *Not-Accessible*/D1 (blind users will find it problematic). It is *Negligible-Cost-per-User*/D2 and *Server-Compatible*/D3 but *Not-Browser-Compatible*/D4 (additional software must be installed). The desktop version is *Quasi-Mature*/D5 (has been implemented for various platforms, has a small user base beyond academic users, and has been formally user-tested but only one variant in small scale). It is *Non-Proprietary*/D6 (no patents are known to the scheme's designers; freely available for download).

The scheme is both *Resilient-to-Physical-Observation*/S1 and *Resilient-to-Targeted-Impersonation*/S2 (an attacker would also need an exact copy of the object file). Since these passwords are essentially random from the viewpoint of an attacker with access to the object file [2], it is also *Resilient-to-Throttled-Guessing*/S3 and *Resilient-to-Unthrottled-Guessing*/S4. Due to the generation involving an essentially random string being salted with a site domain, the scheme is also *Resilient-to-Leaks-from-Other-Verifiers*/S6 and *Resilient-to-Phishing*/S7. The scheme matches Web passwords for the remaining security benefits S8–S11. Notably, S1–S4, S6, and S7 all improve over basic Web passwords. For S8 (*Resilient-to-Theft*), we grant the benefit based on the strict definition, but note that password object files may be stolen if backed up onto portable media to allow device-independence, or made available to attacks through insecure syncing mechanisms (as discussed earlier); here we rate the scheme assuming that neither occurs, and note that, consequently, the scheme has the disadvantage of being device-dependent (which also results in the penalty of not having the benefit *Nothing-to-Carry*/U3).

For the ObPwd mobile version, the following benefits are worth some comments relative to the desktop version: ObPwd mobile has *Quasi-Infrequent-Errors*/ U7, since the ObPwd app requires users to switch to the Gallery app to generate a password, and then paste the password to the requesting Web site. However, these ratings are based on anecdotal comments rather than a formal user study. Regarding the rating *Not-Nothing-to-Carry*/U3 here: syncing mechanisms in the mobile version may also be unavailable, have not yet been proven easy to set up and use by all users with existing mobile phones, or are not used by choice if a secure version is unavailable, in which case a user may need to carry storage media containing their object files. We rate the mobile version as *Not-Mature*/D5 (a version for Android only is available for public download but is more recent than the desktop version, has not been formally user-tested, and has a smaller user base).

The issue of being *Not-Browser-Compatible*/D4 would disappear for ObPwd (as for any other proposal) if the method were widely adopted by browser vendors, and likewise for the missing benefit of being *Not-Mature*/D5, but the ratings measure the status quo, not what could be. Being *Not-Resilient-to-Internal-Observation*/ S5 remains an important drawback, but is strongly related to the existing infrastructure for password verification, a side-effect of aiming to be *Server-Compatible*/D3, which is desirable, at least in the short term, to avoid imposing server-side changes on the password-supporting subset of the approximately 100 million currently active Web sites.

References

[1] K. Bicakci and P. van Oorschot, "A Multi-Word Password Proposal (Gridword) and Exploring Questions about Science in Security Research and Usable Security Evaluation," New Security Paradigms Workshop (NSPW '11), Marin County, CA, USA, Sept. 2011.

[2] R. Biddle, M. Mannan, P. van Oorschot, and T. Whalen, "User Study, Analysis, and Usable Security of Passwords Based on Digital Objects," *IEEE Transactions on Information Forensics and Security (TIFS)* 6(3): 970-979, Sept. 2011.

[3] W. Cheswick, "Rethinking Passwords," Invited Talk at USENIX LISA '10: http://static.usenix.org/events/lisa10/tech/slides/cheswick.pdf. See summary in *;login:* 36(2): 68-69, April 2011.

[4] M. Jakobsson, E. Shi, P. Golle, and R. Chow, "Implicit Authentication for Mobile Devices," USENIX Workshop on Hot Topics in Security (HotSec '09), Montreal, Canada, Aug. 2009.

[5] M. Jakobsson, E. Stolterman, S. Wetzel, and L. Yang, "Love and Authentication," Conference on Human Factors in Computing Systems (CHI '08), Florence, Italy, April 2008.

[6] K. Kostiainen, J.-E. Ekberg, N. Asokan, and A. Rantala, "On-Board Credentials with Open Provisioning," ACM Symposium on Information, Computer and Communications Security (ASIACCS '09), Sydney, Australia, March 2009.

[7] J. Strauss, C. Lesniewski-Laas, J.M. Paluska, B. Ford, R. Morris, and F. Kaashoek, "Device Transparency: A New Model for Mobile Storage," *ACM SIGOPS Operating Systems Review* 44(1), Jan. 2010.