RC23616 (W0503-066) March 10, 2005
Computer Science

IBM Research Report

A Wristwatch-Computer Based Password-Vault

Gabor Blasko
Columbia University
New York, NY

Chandra Narayanaswami, Mandayam Raghunath
IBM Research Division
Thomas J. Watson Research Center
P.O. Box 704
Yorktown Heights, NY 10598

Research Division
Almaden - Austin - Beijing - Haifa - India - T. J. Watson - Tokyo - Zurich

A Wristwatch-Computer Based Password-Vault

Abstract: *Passwords are a popular means of access control to internet services. Unfortunately, during the last few years there has been a proliferation of services that any single individual subscribes to. As confirmed by our survey, users are beginning to rely on assistive methodologies to help them remember their passwords. We argue that a wearable computer with wireless connectivity, processing, input, and display capabilities is well-suited to hold the user's password vault. We describe the design and implementation of such a vault on the WatchPad computer prototype.*

1 Introduction

In the past few years we have seen the widespread adoption of Web technologies to offer the convenience of anywhere, anytime, access to personalized information and services. Browser-based access to services such as banking, bill-paying, stock-trading, shopping, etc., has rapidly grown in popularity amongst customers. Growing customer demand has resulted in a dramatic increase in the availability of such services. Initially some businesses began offering remote access as a competitive edge, but now in several areas, (for instance in banking and bill-paying), remote access has become a must-have feature.

Customers value the ubiquitous availability of Internet based services. It is common for people to access online services not just from their homes and offices, but also from hotel rooms as well as WiFi hotspots that are mushrooming at various places. Since many of these services are based on Web technologies, users also have the convenience of accessing many of these services from any device that supports a web browser.

For each service, the provider typically expects customers to establish a digital identity in form of a user-name and a secret password. The average individual is confronted with the problem of having to remember a large number of user-ids and passwords just to be able to access all the services he wants.

The number and complexity of such access codes and passwords that individuals are expected to remember has been growing steadily over the last few decades with a dramatic increase since the advent of the Web. People working in the engineering and IT sectors have been familiar with access codes for several decades, and have had to deal with the problem of having to remember passwords for access to computing resources. However, for a large fraction of the rest of the population, Web accounts represent their first brush at the problem of having to remember many passwords. A recent Gartner study indicated that 40% of the calls to help desks were related to passwords. Forrester also reported that companies spend $200-$300 per year, per employee to maintain secure passwords

Long before the computer era, proving access rights was usually a matter of producing something one had, such as a door key, or showing an official identification document such as a drivers' license or badge to the person in-charge of verifying credentials. Combination locks offered us the trade-off of carrying the combination in our head versus carrying the key in our pockets. Automated teller machines offered us the convenience of anytime access to cash, provided we could remember a simple 4-digit PIN code and carry the ATM card with us safely. Remote access to phone mailboxes was a convenience that came at the cost of adding one more code to our memory. With the advent of the Internet, individuals have to deal with passwords for everything from accessing their personal computers, internet service provider, corporate VPN, email accounts, bank accounts, retirement accounts, online shopping web sites, gaming sites, instant messaging, frequent flier accounts, mailing list subscriptions, bill-paying services, online photo repositories, etc. In addition, several web sites allow users to personalize the content they see on the site, in exchange for the user remembering a password. Some web sites such as free online newspapers and magazines require users to establish an account with them, even though there is no personalization of the core content that users obtain from these sites.

The problem of having to remember passwords is exacerbated by two aspects that several providers suggest: Having to choose passwords that are difficult to guess and having to change these passwords at regular intervals. Both of these requirements stem from a security perspective, since they prevent unauthorized individuals from getting access to the services by guessing or reverse engineering the passwords. However a complex password that is hard to guess and one that changes often is also hard to remember. To exacerbate the situation users are advised to avoid using the same password for different services. Again while this recommendation arises from a security perspective it is at loggerheads with the limitations of human memory.

Password based authentication schemes are extremely popular for several reasons. First and foremost, a password based scheme is easy to explain to almost all non-technical users who have already been introduced to the concept by ATM pin codes and combination locks. The user instructions are simple: "Choose a secret code and don't reveal it to anyone". Password based schemes are easy to set up remotely and the customer can be granted access instantly at the point the account is set up. In contrast schemes where the provider has to send the customer an object (such as an access control card) are more expensive and may be cumbersome to set up. Password based schemes are relatively easier for the provider to administer and manage. The provider has full control over resetting or revoking lost or compromised passwords. Most importantly current password based schemes can be implemented without having to involve any third party in any of the transactions: account setup, account validation, account administration, or account suspension.

The popularity of password based authentication has already foisted the problem of having to remember passwords upon us, and many individuals are starting to devise their own assistive mechanisms to help them cope with this problem. Commercial products are also becoming available to help users address this problem. Each coping mechanism, either user-devised or a commercial product, represents a certain point in the trade-off between usability and security. Generally assistive mechanisms

that are most easy to use (for instance using the same simple password for all accounts) are also ones that expose the user to the greatest level of vulnerability.

In this paper, we discuss the design of a wrist-watch form-factor device that serves as password vault that also enables a user to follow the recommendations of choosing complex passwords, which are changed often, and are not reused between service providers. We believe that a wearable password vault in a wrist-watch form-factor represents a balance between usability and security that many users will find attractive.

The rest of this paper is structured as follows. In Section 2 we discuss the different approaches that have been used to solve the problem of users having to remember too many passwords. In Section 3 we examine some of the practices adopted by users, both based on the observation of the authors and the results of a user survey that we conducted. We describe the wristwatch computer based password vault we have implemented in Section 4. Section 5 analyses the different password vault approaches and discusses the trade-offs involved.

2 Related Work

Schemes without passwords: If service providers were to adopt client side certificates as an authentication scheme, users could use a single certificate to authenticate themselves to several providers without having to remember passwords. However, client side certificates are difficult for most non-technical users to understand, and very few users actually have client side certificates. A vast majority of users do not understand what a certificate is and how to get one from one of the trusted certificate issuers. Even self-generated PGP style asymmetric keys are beyond the reach of the majority of users [29].

Key fobs such as RSA SecurID can generate time-based pass-codes synchronized with the provider and relieve the user from having to rely on their memory. The user simply looks up the code from the key fob and uses it to obtain access. Some systems require the user to append a small PIN to the code from the fob. Besides being more expensive than a password based scheme, widespread deployment of this solution would require users to carry a large number of key fobs for each provider with whom they have established accounts.

Single password for multiple services: Single sign-on schemes represent an interesting trade-off where a single password enables users to access several different services. This approach works well when a single provider offers several services. For instance a single password enables a customer to access a variety of Yahoo services such as email, instant messaging, online photo repositories, etc. However it is harder for multiple providers to agree to a single sign-on scheme. From a provider's perspective there is a certain degree of control that they have to cede to the enterprise that is managing the single sign-on credentials and many providers choose not to cede that level of control. In addition, single sign-on mechanisms represent an attractive target for hackers. The widespread negative publicity generated by successful break-ins to Passport has slowed the adoption of single sign on schemes by service providers.

Alternative memory based schemes: Another class of solutions that use graphical elements instead of textual passwords has been investigated by several researchers. The underlying philosophy of such approaches is that it is easier to remember graphical elements [14] than text strings. One scheme [Blonder] presents an image to the user and requires the user to select salient features in the image in a particular sequence to gain access. Some researchers [16] have investigated entering a password by drawing a secret. Schemes that require a user to choose a right sequence of images from a large set of images has also been investigated [10,11,15,25]. Variations that scramble the images and vary the duration for which they are displayed are described in [22]. The relative security of various graphical password schemes has been studied in [10, 16, 27]. Approaches to authentication that use gestures created by shaking mobile devices have been investigated in [7, 24]. Camera based authentication schemes are presented in [8].

Password Vaults: One of the popular approaches to dealing with text passwords is to create password vaults that hold passwords on behalf of the user. The simplest of these password vaults which is used by many users is a *pencil and paper* approach where the user keeps a copy of all user ids and passwords on a piece of paper. By keeping it in his wallet, the user has access to his account information in several settings.

Web browsers such as Mozilla and Internet Explorer can automatically fill in password fields in web pages by maintaining password vaults on the computer where they are installed. Some browsers also optionally encrypt the password vault using a master password that the user has to provide.

Small USB keychain fob storage modules such as BioPod™, CryptoGram™, Encentuate™, ActivCard™ can hold an password vaults and supply passwords multiple digital accounts. In order to gain access to the vault on the token the user needs to plug the token into the personal computer and respond to a challenge issued by software running on the computer.

Some USB keychain fob solutions also provide fingerprint scanning capabilities, e.g. ClipDrive Bio™, AuthenTec™, claim to strengthen the user authentication procedure. Typically, a master password and a biometric are used to protect the contents of the USB key. When the user needs to authenticate, he supplies the password and a fresh copy of his finger print. The images are then compared by software on the host computer and if the test passes, the user is given access to contents in the key. Some solutions keep the user's registered fingerprint, master password and the driver software on the key, so that the contents of the key can be used from any computer with an USB interface. More expensive key fobs with onboard processors could perform the fingerprint verification on the key itself and unlock portions of the data on the key.

Wearable tokens with security credentials: Other related work investigates how wearable tokens with security credentials can be used for several purposes. In [9], a file system encryption and decryption system is described using a short-range wireless link between a small non-interactive portable token and a computer. Whenever

the personal computer needs decryption authority, it acquires the decryption authority from the token and the authority is retained as long as necessary. In [1] the authors describe the wristwatch to gain secure access to a space using certificates. In [19] the authors describe a system that allows users who obtain a "wearable ID key" to personalize ubiquitous computers by simply touching them. When users touch ubiquitous computers with their wearable key, the keyholes of the ubiquitous computers recognize their IDs and can personalize the computers. Facile [23] provides a short-range attention correlated channel that gets established between a wearable device and an environmental device. This channel may be used to transfer the password information if available.

The following features distinguish our solution from previous work. First, unlike plain USB keys with no input mechanisms or displays, we use a wearable device to store and supply passwords, as well as generate cryptic passwords if necessary. We also rely on programs running on the wearable device to confirm that it is communicating with a trusted program on a personal computer before releasing the password. Finally, in order to offer better usability across a variety of settings, we allow multiple options – fully automated authentication, semi-automatic authentication, and manual fall-back. In addition we also focus on creating the front-end—an efficient and quick interface—for a wearable password vault, addressing the human-computer interaction issues and system usability aspects. The closest work to ours is by Balfanz [4] who discusses a way to use PDAs which have computing and I/O capabilities to provide better function compared to smart cards.

3 Current User Practices

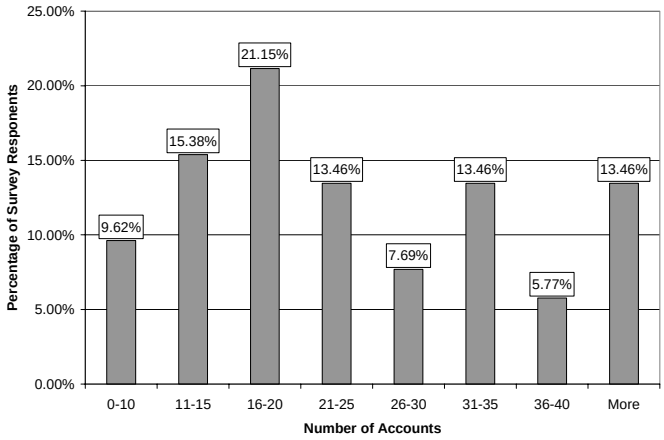
Users sign up for a wide range of different Internet accounts. From a user's perspective, one axis along with accounts can be classified is the frequency with which the user authenticates to the account. Users may authenticate more than once a day to accounts such as email, but they may only log on to a specific internet shopping site. In some cases, such as online newspapers, the provider may simply set a long-term cookie on the user's browser and never prompt the user for a password, until the user tries to access the newspaper from a computer they have not used before. A user may remember that he had registered with the newspaper but not remember the password that was created several months or years ago. Passwords that are used often tend to stay in one's memory while ones that are used infrequently are likely to be forgotten.

Another axis to classify Internet accounts is the perceived value of these accounts on the part of the user. At the low end are accounts that enable users to manage their subscriptions to email mailing lists, or access their free subscriptions to online newspapers or magazines. At the other extreme are accounts which enable users to conduct financial transactions or access sensitive information. If a user is unable to access their low value accounts because they cannot remember the password, the user may be willing to wait for a password reset or simply create a new account with the provider. Users may also not be overly concerned about someone else getting access to their low value accounts by discovering the password used for access. However users

are unwilling to accept unauthorized access to their high value accounts or being unable to access these accounts while they wait for their passwords to be reset.

Unfortunately the classification of accounts based on value is not static. In many cases, when a user first establishes an account with a provider the value of that account may be quite low from the user's perspective, especially when the provider is offering the service for free. Over time, however as the user repeatedly visits the provider and uses the services offered the value of the account tends to grow. The provider may customize the user experience, or the user may create state information with the provider gradually increasing in value over time. The first time one establishes an account with a Web site to view a friend's shared photo album the user may not consider this account a high-value account. However, if the user starts uploading his own photos and sharing them, ordering prints, etc. This account increases in value and the user cannot afford to simply create a new account with the provider.

3.1 User survey

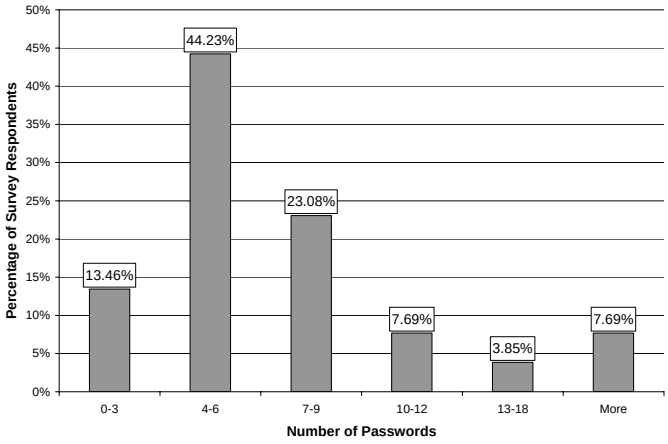


We conducted a user study to confirm our suspicions that Web passwords are overwhelming many users. Previous surveys [2, 17, 20], were largely addressing password practices in the pre-Web era. Our survey was conducted amongst summer interns of a large computer science research institution. We asked users to specify the approximate number of accounts that they need to track. We asked them what assistive mechanisms they were using to help them remember their passwords, if any. We received 52 responses to our electronic questionnaire from computer users in the 18–30yr age range (mean: 24.4yr, std. dev.: 2.8yr). This is a relatively young group of individuals all of whom were well acquainted with computing technology. The questionnaire encouraged freeform answers to most questions. The results are shown in the Figure 1.

Approximately 50% of users said that they have 11–25 online accounts, and around 40% said that they have 26 or more accounts. The mean was 26, ranging from 6–83, with a standard deviation of 14.15. This number seemed surprisingly low based on the experience of the authors. We believe that the actual number of accounts and passwords is significantly higher in reality. When we probed further, many users admitted that they did not create an all encompassing thorough listing of all the passwords they have created. It was interesting to note that respondents had trouble even remembering the various accounts they had created on the internet! Many mentioned that for online web sites they use only infrequently, they depend on the option of having their login information sent to them via email, or having their password reset. 47% of users claimed that in the 30 days prior to the study they needed to have at least one of their passwords reset. Some users also mentioned that they often provide incorrect information when signing up for low-value accounts. And in this case the simple approach of requesting password resets does not work, since they cannot recall the actual information they supplied when signing up for the account. 57% of users admitted that they depend only on their memory to retain their account information, without any kind of electronic or written records, while the remaining 43% mentioned various methods of recordkeeping—storing hints for passwords in an unencrypted file, keeping an archive of account-related emails, and storing unencrypted files on a laptop, a PDA or a cell-phone, or pieces of paper. Two respondents stated that they use encrypted repositories on their laptop or PDA devices.

To help with memorizing their multiple digital authorizations, most users employ various schemes to decrease the amount of actual account information they need to remember. Our survey indicated that users often use the same password for multiple accounts, which according to them fall in similar "security categories". For example, some users said that they use the same password for all electronic newspaper and forum logins, while using significantly different passwords for each electronic banking site. When required by authorization systems to change their passwords at a periodic intervals (e.g., once a quarter), many users said that they employ a small pool of passwords that they permute, or use various primitive puzzles or character alternating algorithms to derive multiple passwords from the same core password (e.g., Dogbert, d0gbert, or Dogbert05). One user mentioned that every month she employs a new theme to change her most frequently used passwords. For example, in the month of the Olympic games, various new passwords would be created by associating sport related words to her account sets, which are categorized and differentiated based on security demands (e.g., web-based newspapers vs. online banking), employing a simple alteration algorithm, which does not change from month-to-month, to derive different passwords.

When asked how many such "core" and unique passwords users have, on average 13% stated that they have 1–3, 44.23% have 4–6, and 23% have 7–9 unique passwords, as shown in Figure 2. Based on these results, we can conclude that one of the most significant problems with current password management practices is that the ratio of the number of accounts/digital identities to the number of unique passwords on average is 4:1. Not surprisingly, users who had a large number of accounts had the worst ratios. The user who had 83 accounts managed them by using 10 unique passwords and their slight alterations, a ratio of 8:1.



It is quite disturbing to see supporting evidence that users so often use just slight alterations of their passwords and consequently multiple accounts can become compromised by cracking just one account (e.g., a primary email account). However, two respondents revealed a less secure practice that can lead to more serious compromises in security. The statements were "I try a few combinations to get the right one." and "I guess my way through." Users may employ such "guesswork" to uncover their account data, when they have not visited a web page for a while and have forgotten which of their small number of user names and passwords they associated to the web page. If a user is trying to login to a website at which they remember they have already registered an account, they will try multiple of their pool of login and password pairs before they gain access, or before they admit that they need to have their login information emailed to them. The more trials a user executes before admitting that they have forgotten the account information, the more password alternates they release to the web site operator.

Therefore, a very threatening conceivable scenario might be for a malicious clone web site, acting as the user's target site using "phished" site content [12], to repeatedly deny the user access to the site and respond with "invalid login" messages, however actually keep a record of all the information entered during the trials. This way a user may unknowingly reveal a multitude of valid digital identities that are used in parallel on other web sites. By being careless, an unsuspecting user may compromise the security of many systems for which they have accounts.

We found that 30.4% of the respondents were unsatisfied and 15.4% were very satisfied with their current system of digital identity management. The remaining 53.8% were fairly satisfied, many commenting that, while they were aware of the insecurity of their approaches, since "it works" and they were "doing fine" or since they haven't found better solutions, they continue to use these practices.

4 Our Solution Design and Implementation

Based on our own personal experiences and the results of our user survey, it is evident that an assistive mechanism is needed to help people remember the various passwords that they have established with various web sites and service providers. One such assistive mechanism is to create a password vault that maintains all of a user's credentials. Since users are unwilling to sacrifice the advantage of ubiquitous access to their Internet services, one of the primary requirements is that the vault be accessible to the user whenever he/she needs it. It is also important that the vault be available from the different locations that a user may want to access their accounts, and it should also work with the different computing devices (laptops, desktops, PDAs, etc) from which the users access their accounts. In addition access to passwords must be quick and easy.

Along with the requirement of anytime, anywhere, access to the vault, comes with the requirement that the contents of the vault be extremely hard to lose. If a user were to lose his/her password vault the user not only gets locked out from all of the services, but an attacker who gains possession of the vault now may have access to all of the user's online resources. The inherently higher value associated with a vault also makes it a target which invites attacks. So it is important that the contents of the vault be kept out of the hands of attackers.

Other desirable features of password vaults include the ability to automatically generate complex passwords that are not based on dictionary words, ability to change passwords periodically as well as avoiding password reuse across different providers.

To meet the objectives of being always accessible and hard to lose, we decided that the right place for the password vault was in a wearable computing device that can be physically attached to the user. To minimize the possibility of misplacing the password vault the user should not have to detach the device, to retrieve passwords. For ease of use, it is preferable that the password vault be able to communicate wirelessly with the computer that is being used to access the online account and provide the required passwords automatically.

In our system, we use the IBM WatchPad [21] wrist watch computer, to hold the password vault. For each account the WatchPad stores the user-id, the password, the URL for the login page, a short descriptive tag that identifies the account and some optional meta-data that is described below. The WatchPad can establish a secure SSL connection using a Bluetooth transport with the other Bluetooth capable devices owned by the user. The password vault in the WatchPad can be used in one of several different modes.

4.1 Usage Modes

When the user is using a known trusted PC such as a home or office computer the password vault provides a greater degree of convenience to the user. The user adds a Bluetooth interface to the trusted PC (if it does not already have one) and installs a custom daemon program on the PC that is able to establish an encrypted channel to the WatchPad. As part of the connection setup, the WatchPad verifies that it is talking

to a trusted daemon program. The daemon process also uses Windows APIs that enable it to communicate with other programs running on the PC and simulate the user typing keystrokes or activating page controls. With such a daemon process, and a short-range wireless link, many of the login processes can be either partially or fully automated, effectively simplifying the operations of retrieving and supplying passwords. When the user wishes to use a computer that either does not have a Bluetooth connection or does not have the daemon program installed, there is a fall-back option where the watch displays the passwords on its screen.

Fully Automated Login System

In the fully automated login system, the user performs a gesture on the touchscreen of the Watchpad to select a particular account that he wants to access. The WatchPad transmits a request to the daemon process on the PC over Bluetooth that includes the user-id and the password for that account. The daemon process then opens a new browser window, loads the appropriate web page, fills out the appropriate login and password fields of the login form by simulating user keystrokes, and sends a mouse event that presses the page's submission button.

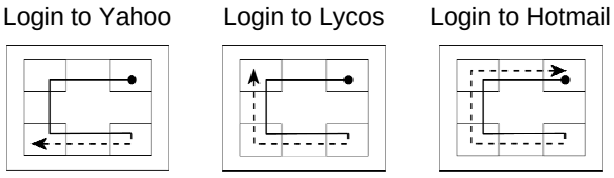


Fig. 3. Examples of concatenated multi-strokes executed on the touchscreen of the WatchPad, used to initiate automated login procedure. First segment *(continuous line)* of stroke invokes command to use the computer watch to send account information. The second segment *(dotted line)*, concatenated to the first to create one continuous multi-stroke gesture is used to specify the web site, the information for which needs to be sent to the web browser on the personal computer.

This fully automated login system is appropriate for a small number of accounts that the user accesses frequently. To automate the login procedure we manually examine the HTML source code to identify the appropriate textbox components that are used by the login script. The URL and the page structure are then registered on the WatchPad as meta-data regarding the account in question. When the user requests a login to that a specific account, the WatchPad sends the meta-data to the daemon. The daemon uses this meta-data find the appropriate text box and button components on the Web page. In order to add a new account to the automated login procedure, or to accommodate the infrequent instances when the provider redesigns the login page, we will need to examine the login page and modify the meta-data regarding the page components.

To simplify the selection of the accounts on the WatchPad, we use a finger gesture on the WatchPad touch screen in the form of a *concatenated multi-stroke* [6]. Examples of such gestures are illustrated in Figure 4. These gestures can be executed by the user eyes-free in less than a second, using tactile guidance, as described in [6]. The entire automated login process takes less than a few seconds, with a significant portion of the lag caused by Bluetooth connection delays.

After executing the concatenated multi-stroke gesture on the watch, the only interaction that we currently ask of the user, is to confirm the execution of the automated login procedure on the desktop computer. This optional confirmation dialog box, only serves the purpose of informing the users during demonstrations of what's about to happen on the PC under the WatchPad's remote control, as not to cause surprises to users who are new to the system. As described above, though this scenario currently depends on manually examining the HTML source of these web pages, we might be able to automate the meta-data generation, if a standard were established to identify the appropriate text entry fields on the login forms.

Partially Automated Login Procedure

If the user wishes to use a trusted PC to login to an account whose web page structure has not been analyzed and is unknown to the WatchPad, the sign on procedure is a bit more complex. The user first manually opens a browser window and navigates to the login page. He then clicks the mouse in the password field of the login form. He then performs a gesture on the watch requesting assistance. The Watch then queries the daemon process for the identity of the active application window, and if the window is a browser window, the URL of the page that is currently open. The URL is used to search through the list of credentials maintained on the watch and the descriptive tags corresponding to the matching entries are presented on the watch face. If the user has multiple accounts with the same Web site there may be more than one entry in the password vault that match that URL, but in most cases only one entry will be found on the watch, simplifying the selection process. Once the appropriate entry has been selected, the watch sends the password to the daemon which enters it into the text field.

For added safety, the Watch verifies with daemon process that the text box where the keyboard cursor is located is indeed a password filed. This step ensures that a user does not inadvertently release a password to some other text box in the login page that can be seen.

Manual Login Procedure

When the user wishes to log in to a web page using a computer which does not have a Bluetooth interface or does not have the host communication daemon installed (e.g. public terminal), the user falls back to the manual login procedure. The initial steps of this procedure are similar to the Partially Automated scheme. The user performs a gesture on the watch to request password assistance. The watch detects that it does not have a secure connection and presents a long list comprising of all the tags for the user to manually scan through and find the tag that describes the password he is looking for. The watch then displays the password string on the watch face for the user to look at and manually transfer to the computer. Again techniques described in [6] can

be used to organize the password store so that the user can quickly retrieve the appropriate password quickly using a small number of touch screen gestures.

4.2 Adding New Login Information to the Watch Repository

Since the watch's input capabilities are limited, with regards to entering alphanumeric data into the digital identity repository, we decided to create a simple application that communicates with the watch and adds new elements to the list of accounts. This is illustrated by actual screenshots taken from the watch and the computer in Figure 5.

When working at a trusted PC, the user performs a gesture on the watch indicating that he wants to add a new entry to the password vault. The Watch communicates with the daemon program that presents a dialog where the user can create a new entry. This dialog allows the user to specify the URL to the login page about to be added. The user specifies a user name and password and a descriptive tag all of which is sent to the Watch for addition to the password vault. A variation of this dialog (not shown) can be used to update existing entries in the password vault.

For improved security we decided to add the ability for users to request the Watch to generate random passwords of various lengths. Since these passwords are not

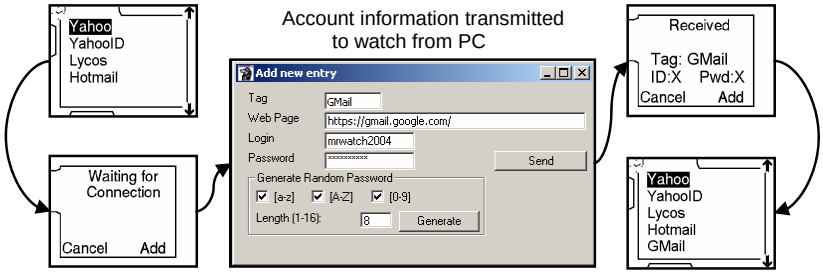


Fig. 4. Screenshots of interface, used to add new entries to the watch computer's digital identity repository.

based on dictionary words they are harder for an attacker to guess. When such auto generated passwords are used in the fully automated or the partially automated modes, the user does not see the actual text string that was auto generated. When used in the manual mode, however, the user will have to correctly transcribe a complex string.

We currently do not support adding new passwords while operating in the manual mode, since it is not very convenient to enter long strings such as URLs etc on the watch.

4.3 Protecting the vault

One additional feature we implemented is a scheme to authenticate the wearer of the WatchPad to the password vault. This authentication scheme is required to guard against the unlikely event of the user losing the watch. In the manual mode the watch displays the password strings on the watch face, so if an attacker were to gain possession of the watch, the attacker could easily recover the contents of the watch with very little effort.

The wearer authentication scheme works as follows. When the user performs a gesture on the watch requesting password assistance, the watch prompts the user to enter a master password on the watch that will open the password vault. The user can

Fig. 5. *(left)* IBM/Citizen WatchPad, displaying pictogram–based authentication challenge interface. *(right)* User can locate tangible corner features of the WatchPad without looking at the device and may execute gestures by dragging the finger along the display's frame, using corner features as starting and stopping landmarks.

configure an inactivity timeout that determines how often the watch will request the master password.

We decided to explore the use of using pictograms as our password alphabet. In [5] the using of color picture password scheme on a PDA is explored. The results of the authors indicate that the human visual memory system is very capable of retaining pictogramic passwords for extended periods of time, and in case the pictograms are constructed from shapes or pictures that are meaningful to the user, they can be easily reconstructed if forgotten. Pictogram passwords are more dependable against over-the-shoulder peeking. While a quick glance may be sufficient to read and remember a simple word or a sequence of digits, it is harder to remember a shape one glanced at, let alone remember a sequence of shapes.

Since the display area is small we decided to use an interface solution, which we call *content cards*, to expand the screen area. The way content cards work is illustrated in Figure 6. A total of 32 pictograms are placed on virtual content cards which may be pulled into the main screen area by executing dragging strokes from one corner of the touchscreen to the neighboring corner. Since there are four screen corners

and gestures may be executed in either of two directions (clockwise or counterclockwise) the total number of such content cards is eight. When one of these cards is dragged in the four pictograms contained on the card are displayed. At this point the user lifts his finger off the touchscreen to see the pictograms and selects one by tapping in the appropriate quadrant of the touchscreen. By following a similar sequence of drags and taps the user enters a master password consisting of pictograms. Expert users who have memorized the positions of the pictograms in their password can use concatenated multi-strokes [3?] to quickly enter the pictogram password without looking at the watch face at all. If the user does not correctly respond to the wearer authentication challenge, the watch refuses to release any passwords. After a certain number of incorrect authentication responses the watch should erase the password vault. (This feature is not yet implemented).

We currently use a master password consisting of four pictograms, each pictogram chosen from the 32 possible ones. Clearly, this password mechanism is only equivalent to using a 20 bit key, and is not very strong. It is only intended to make it difficult for an unsophisticated attacker who has possession of the watch. A determined attacker with physical possession of the watch has other means to extract the passwords from the watch, such as probing the hardware to dump the contents of memory. We believe that this trade-off between usability and security is appropriate given that low probability of losing a wristwatch.

Currently the passwords are stored in plain text form on the watch in Flash memory. A simple enhancement we propose to add is to store them in an encrypted fashion, where the encryption key is derived from the master password that the user enters. (i.e., the encryption key cannot be generated from the contents on the watch). Once the wearer authenticates the password store is copied into the DRAM and decrypted. When the wearer authentication times out, or power is interrupted the DRAM copy is erased. With this enhancement, an attacker who dumps out the Flash memory contents of the watch will have to go through the extra step of decrypting the contents of the memory. Unless we also increase the length of the master password, this level of encryption is unlikely to deter a sophisticated attacker who has the ability physically open the watch and dump out the memory contents.

An alternative design that increases the length of the master key is to use a scheme similar to physical combination locks, where the user selects a sequence of numbers using alternating clockwise and counter clockwise movements. The movement can be controlled using the roller wheel and an arrow can be moved on the screen to provide feedback. With this approach the “alphabet” increases to 60, and if we use a sequence of 5 numbers, we get an effective key length of about 29 bits. It does not seem likely that we will be able to generate a 256 or 512-bit key very easily using the input mechanisms available on the watch without seriously impacting usability.

A further enhancement to the watch would be to add a sensor to the buckle or strap to can detect when the watch is being worn and when it is being removed. This sensor can increase the usability by posing the wearer challenge only when the wearer puts on the watch instead of using a timeout mechanism.

Our watch also includes a fingerprint sensor which can be used to authenticate the wearer. At present, however, we do not have fingerprint recognition software on the watch.

5 Analysis

We can classify the different password vault mechanisms based on where the vault is stored and how it is used. This classification is shown in Table 1 below. Manual mode vaults require the user to look up the password from the vault and manually enter the password into the Web form, while automatic vaults fill in the passwords for the user automatically. All of the approaches where a user maintains his personal vault are generally manual mode vaults unless the user has built the automation software.

Table 1 : Classification of Password Vaults

Vault Location	**Manual mode**	**Automatic mode**
Offline	Paper-pencil	N/A
In PC	User managed file	Browser Vault
In portable token	User managed file	USB Vault
In wearable device	WatchPad manual mode	WatchPad fully/partially automated modes

Manual mode operation, with or without the use of password vaults is susceptible to "phishing" attacks[]. An attacker sets up a site that looks similar to the legitimate site. If a user were to end up at the attacker's site due to a typographic error or following a malicious link, the user is fooled into thinking that he is authenticating himself with the site he visually perceives. The attacker captures the password and redirects the user to the actual site. Automatic vaults can check the domain name and the URL to ensure that they match and will not be fooled by the visual similarity of the attacker's web site.

The different password vault approaches exhibit different trade-offs between usability and security as shown in Table 2 below.

Table 2 : Trade-offs based on Password Vault location

Usability & Security Issues	**Manual Vault carried on person**	**Vault in PC**	**USB Vault**	**Wearable Vault (WatchPad)**
Available on multiple machines	Yes	No	Yes, if user remembers to take it with him	Yes
Availability on borrowed PCs	Yes	No	Not recommended	Manual mode only
Ability to encrypt vault	Difficult	Yes	Yes	Yes
Ease of updating passwords in vault	Difficult	Needs replica management	Easy	Easy on trusted PCs

Accidental loss	Less likely	Unlikely	Likely	Unlikely
Cost of solution	Low	Low	Low-Medium	Medium
Battery dependency	None	None	None	Yes
Notification/control over individual password releases	Yes	Possibly	Possibly	Yes
Defense against attacker gaining physical control of vault	None	Password based encryption	Password based encryption or biometric	Weak password encryption or biometric
Trojan on PC being used	Loss of passwords used	Loss of entire vault	Loss of entire vault	Loss of passwords used

User managed manual vaults (e.g. cheat-sheet in wallet) are a low cost solution that is extremely common. While the user has the advantage of being able to access accounts from any computer, including public PCs at airports, a list of passwords kept on a piece of paper cannot be easily encrypted. One way of "encrypting" the passwords is to use mnemonics rather than the actual passwords written in clear text. A second drawback of this method is that when passwords are to be updated, the user has to erase the entry delicately to avoid accidentally erasing other nearby entries because the entries are written in tiny font so that the cheat sheet is physically small. Moreover, after a few write-erase cycles the cheat sheet develops becomes unusable and the user has to copy all his entries to a new cheat sheet.

Browser managed vaults that are stored on one machine are popular among several users, but their drawback is that they are not available to the user when using a different machine. Some users have also created their own personal password vaults in the form of files they store on their computers, perhaps in an encrypted form. Password management software products also exist, Roboform, PassCrypt, AccountLogon, etc., that provide an intuitive user interfaces to vaults stored on a single computer. Maintaining separate vaults on the different computers is problematic because the vaults may get out of sync when the user signs up for new accounts or changes passwords with a provider. The password vault itself is an attractive target for attackers who may seek to make a copy of the vault and try to decrypt it offline [13].

USB storage based vaults can potentially be available on multiple machines, but one major practical problem with USB key tokens is that the user may forget to take the token with him when he leaves the PC. It is difficult to remember to take the key out every time because the user is in a different mental frame when he is leaving and there are no conscious cues that remind him to do so. Leaving behind the vault essentially means that the user no longer has access to his passwords. Worse yet, if the vault is left at an insecure location it may fall into the wrong hands.

The contents in the browser vaults or USB vaults can be protected using passwords that the user enters on the PC, which can be longer than the ones that can be entered on a wearable device. Once these passwords are entered on the PC, a Trojan program

on the PC could potentially steal all of the passwords from the vault and send to an attacker. Even with biometric authentication USB tokens are vulnerable to such an attack. One drawback with biometric techniques is that they are hard to revoke if a copy of the biometric is stolen [26].

6 Conclusions

Password vaults are a coping mechanism that helps users deal with the proliferation of user accounts in a web-based world. In this paper we argued that a good place to store the vault is on a wearable computer with a display and user interface capabilities. We have implemented a solution using the WatchPad prototype. Our solution offers different modes of operation to meet the user's needs of ubiquitous access to online accounts. We show that a well designed user interface to a wearable password vault offers high usability while ensuring a reasonable level of security. We have also compared existing solutions and characterized them based on the trade-offs they represent. We believe that a marriage between password vaults and wearable computers is a practical solution until we devise usable authentication schemes that respect the limits of human memory.

References

1. Al-Muhtadi, J., Mickunas, D., and Campbell, R., "Wearable Security Services," in *Proceedings 21st International Conference on Distributed Computing Systems Workshops*, 2001, pp. 266-271.
2. Adams, A. and Sasse, M. A., Users are not the enemy: Why users compromise computer security mechanisms and how to take remedial measures. *Communications of the ACM*, 42(12):40-46, December 1999.
3. Anderson, R.J., Why Cryptosystems Fail. *Communications of the ACM*, 37(11):32-40, November 1994.
4. Balfanz, D., and Felten. E., "Hand-Held Computers Can Be Better Smart Cards," *Proc. Usenix Security 99*, , Usenix, 1999, pp. 15-24.
5. Blonder, G.E., Graphical password. *US Patent 5559961*, August 30, 1995.
6. Blaskó, G., Feiner, S.: An Interaction System for Watch Computers Using Tactile Guidance and Bidirectional Segmented Strokes, In *Proceedings of 8th IEEE International Symposium on Wearable Computers*, 2004, Arlington, VA, USA
7. Castelluccia, C, and Mutaf, P, - Shake Them Up! A movement-based pairing protocol for CPU-constrained devices Rapport de recherche de l'INRIA RR-5457 - Jan 2005.
8. Clarke, D., et al., "The Untrusted Computer Problem and Camera-Based Authentication," *Proc. Pervasive 2002*, Springer-Verlag, 2002, pp. 114-124.
9. Corner, M. D., Noble, B. D.: Zero-interaction authentication. In: *Proceedings of the 8th Annual International Conference on Mobile Computing and Networking*, pp. 1–11, 2002.
10. Davis, D., Monrose, F., Reiter, M.K, On User Choice in Graphical Passwords, In *Proceedings of Usenix Security Symposium*, 2004, pp. 151-164.
11. Dhamija, R. and Perrig, R., "DejaVu: A User Study Using Images for Authentication," *Proc. 9th Usenix Security Symp.*, , Usenix, Aug. 2000, pp. 45-58.

12. FTC Consumer Alert: How Not to Get Hooked by a 'Phishing' Scam: http://www.ftc.gov/bcp/conline/pubs/alerts/phishingalrt.htm
13. Gutmann, P., How to recover private keys for Microsoft Internet Explorer, Internet Information Server, Outlook Express, and many others - or - Where do your encryption keys want to go today? *http://www.cs.auckland.ac.nz/~pgut001/pubs/breakms.txt*, 1997.
14. Haber, R.N. How we remember what we see. *Scientific American*, 222(5):104-112, May 1970.
15. Jansen, W., Gavrila, S., Korolev, V., Ayers, R., Swanstrom, R.: Picture Password: A Visual Login Technique for Mobile Devices, National Institute of Standards and Technology Interagency Report 7030, July 2003.
16. Jermyn, I., Mayer, A., Monrose, F., Reiter, M. K., and Rubin, A.D., The design and analysis of graphical passwords. In *Proceedings of the 8th USENIX Security Symposium*, August 1999.
17. Klein, D. V.; "Foiling the Cracker; A Survey of, and Improvements to Unix Password Security", In *Proceedings of the 14th DoE Computer Security Group*, May 1991.
18. Matsumiya, K., Aoki, S., Murase, M., Tokuda, H.: Zero-stop Authentication: Sensor-based Real-time Authentication System, In: *Proc. 9th Real-Time, and Embedded Computing Systems and Applications* (*RTCSA 2003*), 2003 pp.179–194,
19. Matsushita, N., Tajima, S., Ayatsuka, Y., Rekimoto, J.: Wearable Key: Device for Personalizing Nearby Environment. In *Proceedings of ISWC* 2000: 119-126
20. Morris, R., and Thompson, K. Password security: A case history. *Communications of the ACM*, 22(11), Nov 1979.
21. Narayanaswami, C., et al. IBM's Linux Watch: The Challenge of Miniaturization, *IEEE Computer*, Jan 2002, pp 33-41.
22. Narayanaswami, C., Password protection using spatial and temporal variation in a high-resolution touch sensitive display, *US Patent 6,720,860*, April 13, 2004.
23. Partridge, K., Newman, S., Borriello, G.: Facile: A Framework for Attention-Correlated Local Communication. *WMCSA 2003*.
24. Patel, S.N., Jeffrey S. Pierce, J. S., Abowd, G.D.: A gesture-based authentication scheme for untrusted public terminals. *UIST 2004*, pp. 157-160.
25. Pering, T., Sundar, M., Light, J., Want, R., Photographic Authentication through Untrusted Terminals, *IEEE Pervasive Computing*, Vol 2, No. 1, pp 30-36.
26. Ratha, N., Connell, J., Bolle, R. Cancelable Biometrics In *Proceedings 2000 Biometrics Consortium Workshop*, September 2000
27. Thorpe, J., van Oorschot, P., Towards Secure Design Choices For Implementing Graphical Passwords, *20th Annual Computer Security Applications Conference*, 2004, pp. 6-10.
28. Whitehouse, O. Where Security and Business intersect, *CanSecWes*t, April 21-23, 2004, http://cansecwest.com/csw04/csw04-Whitehouse.pdf
29. Whitten, A., and Tygar J. D. Why Johnny can't encrypt: A usability evaluation of PGP 5.0. In *Proceedings of the 8th USENIX Security Symposium*, August 1999.