WebTicket: Account Management Using Printable Tokens

Eiji Hayashi[1] **Bryan A. Pendleton**[1] **Fatih Kursat Ozenc**[2] **Jason I. Hong**[1]

[1]Carnegie Mellon University, 5000 Forbes Ave. Pittsburgh PA 15213, USA,
{ehayashi, bpendlet, jasonh}@cs.cmu.edu
[2]Autodesk Inc. 1560 Trapelo Road, Waltham, MA 02451, USA, kursat@cmu.edu

ABSTRACT
Passwords are the most common authentication scheme today. However, it is difficult for people to memorize strong passwords, such as random sequences of characters. Additionally, passwords do not provide protection against phishing attacks. This paper introduces WebTicket, a low cost, easy-to-use and reliable web account management system that uses "tickets", which are tokens that contain a two-dimensional barcode that can be printed or stored on smartphones. Users can log into accounts by presenting the barcodes to webcams connected to computers. Through two lab studies and one field study consisting of 59 participants in total, we found that WebTicket can provide reliable authentication and phishing resilience.

Author Keywords
Password; User Authentication; Usable Security

ACM Classification Keywords
H.5.2 Information Interfaces and Presentation: Miscellaneous

General Terms
Human Factors and Security

INTRODUCTION
Passwords are the most commonly used form of authentication for web services today. A fundamental assumption here is that users can memorize secure passwords. If users have only a few passwords, it is possible to memorize them. However, as the number of passwords increases, users have difficulty in remembering them, in part due to interference effects [4].

The cost of forgetting passwords is not trivial. For the NYTimes online, 100,000 readers forget their passwords each week. Furthermore, 15% of *new* readers were actually old readers signing up again because of a forgotten password [13]. As another example, a Gartner report investigated the cost of forgotten passwords at a large beverage company. This investigation found that 30% of help desk calls were related to passwords, with an average cost of $17.23 USD per call, resulting in an annual impact of more than $900,000 USD [11].

CHI'12, May 5–10, 2012, Austin, Texas, USA.

(a) Front (b) Back

Figure 1. An example of a paper-based WebTicket. On its front side, it has a two-dimensional barcode that stores a login script. On its reverse side, it has a favicon, user-editable text, a user-chosen border, and instructions.

People cope with the burden of passwords in many ways, but these strategies often lead to new security risks. For example, to improve memorability, users often choose *simple, easy-to-remember passwords* [4,28]; however these simple passwords are highly vulnerable to dictionary attacks [16,18] and educated guess attacks. Furthermore, an analysis of 32M passwords exposed in a security breach at RockYou.com showed that the top 20 most common passwords could compromise over 5% of the accounts [33].

Another strategy is to *reuse passwords*. Gaw and Felten [12] and Hayashi and Hong [15] both reported that users often reused passwords. Reusing passwords reduces the number of passwords that users must memorize; however, if one account using a shared password is cracked, other accounts are potentially compromised.

In this paper, we introduce WebTicket, a web account management system that lets users manage their accounts without memorizing passwords. WebTicket generates a strong password and embeds a login script – which includes a URL of a web site, a user ID, and the password – in a 2D barcode on a ticket. Users can print tickets (see Figure 1) or store tickets on their mobile phone (see Figure 4). To log into an account, users show the ticket to a webcam on their computers. Data is encrypted using a key stored in the user's computer. Thus, an attacker must have access to both the user's computer and the ticket.

WebTicket transforms knowledge-based authentication into token-based authentication at a relatively low monetary and operational cost. WebTicket also prevents phishing attacks by steering users to legitimate websites. Additionally, even when led to phishing websites, users cannot type in their passwords because they do not know their own passwords.

WebTicket is not intended to replace all passwords. WebTicket works well in managing infrequently used or secure passwords that users are more likely to forget, while passwords work well for frequently used accounts where users are less likely to forget passwords.

In this paper, we present the design and implementation of WebTicket. We also present the results of two lab studies and one field study consisting of 59 participants in total, which suggest that WebTicket can provide easy and reliable user authentication while being resilient against phishing.

RELATED WORK

We have organized related work into two categories: usable security and tangible interfaces.

Usable security

User authentication systems depend on three types of mechanisms: *what you know*, *what you have* and *what you are*. In *what you know*, an authentication system and a user share a secret at enrollment, e.g., a password. The system authenticates the user by verifying whether the user knows the shared secret. In *what you have*, a system authenticates a user based on whether the user has a physical object, such as a credit card, which is given to the user at enrollment. In *what you are*, an authentication system records some aspects of a user's physiology or behavior at enrollment, and, then, authenticates the user based on whether this property matches or not.

Password managers. Much past work has examined how to reduce memory workload to make account management for *what you know* authentication easier. The most straightforward approach is to store user IDs and passwords in computers. All major web browsers have a built-in password manager. However, if attackers have access to a user's web browser, they can access the user's accounts without any authentication. Furthermore, if users want to use the account information on multiple computers, users have to synchronize the computers whenever they create or modify the account information (e.g., changing passwords). The synchronization, in many systems, is done by storing the information in centralized servers, which can be targeted by attackers because it contains information about many accounts from multiple users. Moreover, users have to trust third parties to do the synchronization.

A common alternative to password managers is to simply write down passwords on paper. In this approach, once attackers obtain access to written passwords, the attackers could compromise the accounts by guessing their user IDs, which are e-mail addresses in many cases, and trying some popular websites. Moreover, writing downs passwords does not provide any phishing resilience.

Another possible approach is that users reset passwords whenever they want to log into their accounts via e-mails or by answering secret questions. However, resetting passwords could be time-consuming processes.

Master password approach. Systems such as PwdHash [26] and PassPet [32] decrease the number of passwords that users have to memorize by generating account specific passwords based on a single master password. In these schemes, users only need to memorize one password; however, these schemes also rely on additional information (i.e., domain names or user-chosen name for web sites) that is easy to guess or obtain. For example, attackers could launch online attacks on insecure web sites, which do not restrict number of trials, to obtain a master password, then, compromise secure and important accounts.

Mnemonic passwords. Another approach to making passwords easier to memorize is to use mnemonic passwords, which are seemingly random sequence of letters generated from a phrase. For instance, "Ilts@7S!" can be generated from a phrase "I love to ski at Seven Springs!" Mnemonic passwords are easy for users to memorize and relatively difficult to guess for attackers [31]. However, because users are likely to choose specific phrases as sources of mnemonic passwords, attackers can guess mnemonic passwords more easily than random passwords [18]. Furthermore, even using mnemonic passwords, users can forget which password is for which account, due to scaling issues and interference effects.

Graphical passwords. Another solution to making passwords easy to memorize is using graphical passwords [5,14,30]. Graphical passwords are based on the observations that people are better at memorizing (or recognizing) graphics than at memorizing text [19,25]. However, these graphical password authentication schemes have challenges in actual deployment because of uncertainty about their security and scalability [6,8,10].

Tokens. Other authentication systems depend on *what you have*. eToken is a USB device that can be used as a "physical key" to login [1]. A one-time password token is a device with an LCD, which shows numbers based on the current time and a key stored in the token. To authenticate, users can type the number shown on the device. Then, a server-side application verifies whether that number was actually generated by that device. RSA securID is a variant of the one-time password token. In addition to a number shown on the RSA securID, users have to type their personal identification number to be authenticated [3]. However, there are challenges in scaling these kinds of approaches across all accounts a person has, since these tokens require server-side support, and it would be impractical to carry a custom token for each web site. As such, these tokens tend to be used only for accounts with very high security requirements.

Finally, there are systems that use smartphones for authentication. Phoolproof Phishing is an authentication scheme designed to prevent phishing attacks, key loggers, and other kinds of malware [26]. Users select what site to login to on their mobile phone, which opens the web site on a local computer. The mobile phone also checks the site's certificate to verify that the opened site is the correct site, at which point users can login normally. While WebTicket does not offer as strong mutual authentication, it does not

require server-side changes, and also facilitates logins by entering usernames and passwords. Seeing-Is-Believing is a pairing protocol for cell phones [21]. The protocol pairs two devices by conveying a hash value, i.e., a number, using a 2D barcode. In contrast, in WebTicket, a 2D barcode on a ticket conveys richer information, such as a URL, a user ID and a password for user authentication.

Tangible Interfaces

There are many tangible interfaces that make use of paper (e.g., [17,20,22]). Palette is a presentation tool that lets presenters access digital presentations quickly using physical cards with printed barcodes [24]. Collaborage is an augmented board where users can put paper tags with 2D identification codes, which connect the paper tags with information on a computer [23]. These projects demonstrated that users could manage complicated information well using paper augmented by computers.

WEBTICKET DESIGN GOALS

Our primary goal with WebTicket is to support people in accessing infrequently used accounts as well as accounts with stringent password requirements. WebTicket is also intended to help novice users who are uncomfortable with computers or have a hard time remembering not only their passwords, but also what website to go to. In this sense, WebTicket can be thought of as a tangible web bookmark. Towards these ends, we defined the following design goals.

Be Reliable for Logins. WebTicket should support logging into accounts reliably even after a long period of inactivity.

Be Easy to Understand. Users often have difficulties in understanding security systems because of a lack of awareness and knowledge about computer security [29]. We wanted WebTicket to offer users a simple mental model of how it works and how to use it, which should help to improve security in practice.

Offer Strong Passwords. Users tend to choose easily guessable passwords or reuse passwords to improve memorability [4,31]. To maximize the security provided by passwords, WebTicket should use a randomly generated strong password for each account.

Be Compatible with Existing Web Sites. Virtually all web sites today use password-based authentication. We want WebTicket to be compatible with the majority of web sites without any server-side modifications.

Be Low Cost. Some account management systems entail additional costs, in terms of initial purchase costs, setup, and maintenance, which can be barriers to adoption. Hence, we wanted these costs for WebTicket to be low.

Support Partial Adoption. Rather than forcing people to switch completely to a new system, we wanted to let people choose which websites they wanted to use WebTicket for.

WEBTICKET DESIGN AND USAGE

Using the WebTicket browser extension, a user can create a ticket with a 2D barcode that contains a login script, and then print the ticket or store the ticket on their smartphone. This login script includes the URL of a web site, the user ID, and the encrypted password. To log into the account, the user shows the ticket to a webcam attached to her computer. WebTicket scans the 2D barcode, opens the website in a browser, and logs into the account, without requiring the user to enter in her user name and password.

In this section, we provide more detail on installation, account creation, tickets, and the login process (See Figures 1 to 3), along with our design rationale. Near the end of the paper, we discuss alternative designs and limitations.

Installing WebTicket

We implemented the backend of WebTicket as a Firefox extension. On installation, WebTicket generates a random number that is used as a key to encrypt information on tickets paired with that computer. Users can print a backup of the encryption key as a special kind of WebTicket if desired, which lets users copy the key to other computers. After copying the keys, tickets generated on one computer can be used on other computers as well without any synchronization. If the backup is printed, the backup should be kept securely to ensure the security of the encryption.

After generating the key, WebTicket prompts users to choose a border from a large set of borders, which is printed on tickets (see Figure 1.b) and displayed on the WebTicket reader dialog (Figure 3). This border is based on dynamic security skins [7]. Because the border is a secret shared between a user and the WebTicket system, an attacker will have a harder time creating a fake phishing dialog that pretends to be WebTicket. Note that the border is an additional layer of security against phishing attacks. Even if an attacker obtains the 2D barcode on a ticket by letting the user scan her tickets using a fake reader dialog, the attacker cannot recover account information without the encryption key stored in her computer(s). WebTicket included dynamic security skins as a form of defense in depth.

WebTicket Account Creation

When signing up for web services, users create web accounts and tickets using the WebTicket wizard (Figure 2). The account creation process consists of (1) *signing up*, (2) *recording*, and (3) *printing*. In the *signing up* step, a user is asked to sign up for the web service, e.g., amazon.com. WebTicket automatically fills password fields with a randomly generated strong password consisting of upper case, lower case, numbers, and at least one symbol.

One challenge in generating passwords is that different web sites have different password policies. We analyzed the 100 most frequently visited web sites listed by Alexa. Of these, 42 allow free sign up. 78.6 % of the 42 web sites were compatible with our password generation algorithm. We think the coverage was reasonably high as an un-optimized research prototype. Other web sites had password policies not compatible with our algorithm (e.g., do not allow symbols). In these cases, users can type letters in password fields to satisfy the policies. WebTicket then randomizes the order of letters to make the passwords stronger.

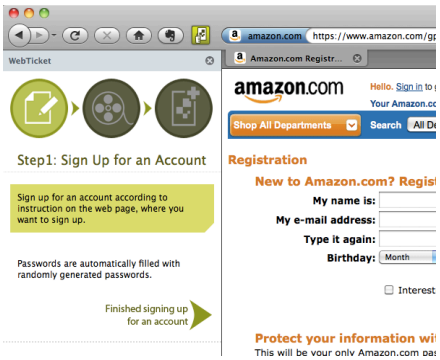

Figure 2. WebTicket wizard. Users can open the wizard from the "create ticket" tab in WebTicket's reader dialog (Figure 3), and then create an account and a ticket using this wizard.

In the *recording* step, the wizard asks the user to log into the web service. When the wizard detects that the user is typing in their user ID, the wizard fills in the password field with the password generated in the previous step. Then, when the user clicks the login button, the wizard records the login process of the web site, i.e., the URL of the login page, the user ID, the password, and names of relevant HTML objects. We currently focus on web sites where users can log into their accounts in a single step, e.g., typing a user ID and a password, and then clicking a login button. Technically, WebTicket could support more complicated login processes, such as those that have intermediate steps like showing a picture, as long as the process requires same input every time, though we do not currently support this.

In the *printing* step, the user can edit text that appears on the reverse side of the ticket to personalize a ticket with a name or a short memo before printing the ticket.

WebTicket also supports creating tickets for existing accounts. Users can create tickets using existing passwords, or can manually change a web site's password to one generated by WebTicket.

The Ticket

Each ticket is 52mm×85mm. A QR Code on the front side contains a login script. QR Codes are 2D barcodes standardized by Denso [2]. We choose QR Codes because they can be easily decoded by a webcam. However, any barcode that can store enough data can be used.

The password in this script is encrypted using the concatenation of a key stored on a user's computer and the URL of the associated web site. The key prevents simple theft or camera-based attacks. The reverse side of a ticket has a favicon of the web site associated with this ticket, user-editable text (e.g., the user's name), the user-chosen border, and instructions about the login process. Instructions were added after the first user study to help users to remember the log in process of WebTicket.

If a ticket is lost, stolen, or damaged, users can revoke the ticket and print a new ticket. Because a ticket is tied to a combination of a URL, a user ID and a password, users can revoke a ticket by changing the old password to a new one using password reset mechanisms provided by web sites. Then, users can print a new ticket that contains the new password using the WebTicket wizard.

Because tickets are physical objects, we can describe tickets as being analogous to physical keys. This description could help users to foster a better mental model of how they should treat their tickets. Furthermore, because a ticket is associated with one account, users can manage them differently. For instance, users can keep tickets for their bank accounts securely at their home, while carrying around tickets for e-commerce web sites.

Logging in Using WebTicket

To log into an account, the user first click on the WebTicket icon in Firefox's toolbar to launch WebTicket's reader dialog (Figure 3). As noted previously, the border of this dialog is identical to the border selected by the user on installation, as an extra protection against phishing attacks [7] in addition to the encryption of the barcode. The dialog also shows a real-time image captured by the computer's webcam. After launching the dialog, the user should check whether the border on the ticket matches the border on the dialog, to avoid custom phishing attacks on WebTicket. If the user sees that the borders match, the user scans the QR Code on her ticket. The WebTicket application then decrypts and executes the script to log into the account.

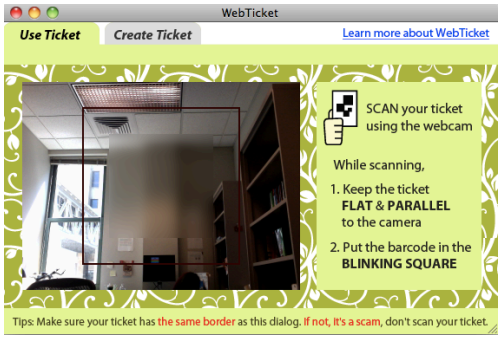

Figure 3. WebTicket's reader dialog. Users scan their tickets by showing their ticket to a webcam. The user-chosen border is shown around the dialog. The color of the green part in the dialog also changes along with the border.

(a) List View (b) Information (c) QR Code

Figure 4. Screenshots of smartphone-based WebTicket. All the tickets in a phone are listed in (a) List View. When user chooses one of them, (b) information is displayed. (c) QR code is displayed when the *show barcode* button is clicked.

WebTicket on Mobile Phones

In the user study #1, some participants expressed concerns about carrying tickets. We also realized that there were potential problems with managing large number of tickets.

To address these concerns, we implemented WebTicket for Android smart phones (Figure 4). Instead of printing tickets, users can use their mobile phones to scan a QR code shown on a computer's display to import the data in *printing* step. To log into an account, a user can select an account on their smart phone to display a QR code. Because the paper-based and mobile-phone version of WebTicket use the same QR code, they can be used interchangeably.

USER STUDY #1: BASIC USABILITY

We conducted a user study consisting of two sessions to compare WebTicket with passwords in terms of their usability using a within-subject design. In the *password condition*, participants created accounts and logged into the accounts using passwords. In the *WebTicket condition*, they did the same using the paper-based version of WebTicket. At this point, we had not implemented mobile phone-based WebTicket. We used three mock e-commerce web sites (for a practice and the two conditions). The websites were different in content and color while they were identical in terms of registration and login forms. The login form, shown on their home pages, required participants' email addresses and passwords to log into their accounts. We designed these web sites based on an analysis of 27 existing e-commerce sites.

Participants

We recruited 20 participants using an existing university recruitment site. Their ages ranged from 21 to 57 with a mean age of 31.9 (σ=11.5). Nine participants were males and 11 were female. They consisted of 11 university students, nine university staffs and two domestic residents. None of the students were computer science majors. We paid $10 USD for their participation and an additional $3 USD for each successful login in the second session.

Procedure

In the initial session, we first asked participants to create an account and then log into that account using WebTicket as a practice trial. After that, for the actual study, we asked participants to create two accounts, one using WebTicket and one using standard passwords in randomized orders. In both conditions, they started from a blank page opened in Firefox, with the WebTicket wizard opened in the WebTicket condition. For the password condition, we asked them not to use their existing passwords. These were intended to improve the internal validity of this study.

After creating the accounts, we asked the participants to log into the accounts starting from a blank page. In the password condition, the participants opened a web site associated with their account using a bookmark. Then, the participant typed their email addresses and passwords in a login form to log into their accounts. In the WebTicket condition, the login process involved more steps. We prepared nine extra tickets for the e-commerce web sites. Then, a participant's ticket was shuffled in with the nine extra tickets to imitate a situation with 10 WebTicket accounts. This is based on the observations that users have about 10 accounts on average [12,15]. In addition, we asked each participant to put the stack of 10 tickets in a place where she kept other cards, e.g. a wallet in a bag. We asked participants to start the login process from this state. Therefore, a typical login process involved taking out a wallet from a bag, taking out the stack of tickets from a wallet, searching a ticket for the specified web account from the stack, launching the WebTicket reader dialog, and scanning the ticket. In both conditions, we regarded a login process as successful if participants could log into their accounts within two minutes. At the end of the initial session, we asked participants to complete a survey. We also asked them to keep their tickets for one week and bring them back to the follow-up session.

One week later, in the follow-up session, we asked the participants to log into their accounts using passwords and tickets. We also asked them to complete another survey.

We conducted this study in an isolated room. An experimenter sat next to a participant and gave instructions. The same experimenter supervised all participants. We used a Macbook Pro laptop (MB766LL/A) with a built-in camera, and a Canon iP4200 printer. We placed the printer and a pair of scissors next to the laptop. We videotaped the whole user study using two camcorders for analysis.

RESULTS OF USER STUDY #1

Account Creation

Table 1 shows account creation times and their breakdowns. In total, creating an account in the WebTicket condition took 71.1 seconds longer than that of the password condition. In the WebTicket condition, it took less time for participants to complete registration than in the password condition. This was because in the WebTicket condition, the wizard generated passwords on behalf of the participants, while they had to come up with passwords by themselves in the password condition.

	Password	WebTicket
(1) Go To a registration page	20.0 (17.1)	15.9 (8.9)
(2) Complete registration	*73.0 (33.8)	*50.0 (18.7)
(3) Record login process	-	34.4 (21.3)
(4) Edit text on a reverse side	-	4.8 (2.8)
(5) Print a ticket	-	19.7 (0.7)
(6) Cut and fold a ticket	-	39.3 (11.2)
Total	*93.0 (45.8)	*164.1 (38.3)

Table 1. Breakdowns (means and standard deviations) of the account creation times in seconds. Asterisks denote statistically significant differences between the password and the WebTicket conditions ($p < 0.05$ in Mann-Whitney U test).

Login Process

Table 2 shows the overall success rates. On the first day, all participants succeeded in both conditions. However, one week after, five participants could not log into their accounts in the password condition, while all participants brought back their tickets and logged into their accounts in the WebTicket condition. Additionally, among those who succeeded in the password condition, it took an average of 1.93 login attempts before successfully logging in. This result indicated that participants were likely to forget their passwords, while they could keep a ticket for one week.

Table 3 shows the login times of participants who successfully logged in. On the first day, the mean time for the password condition was shorter than the WebTicket condition ($p<0.05$ using Mann-Whitney U test). However, one week later, the login time for the password condition increased while that of the WebTicket condition did not, leading to there being no statistically significant difference in the mean login times between the password and the WebTicket conditions.

	On the first day	One week later
Password	100% (20/20)	75%* (15/20)
WebTicket	100% (20/20)	100%* (20/20)

Table 2. Success rates of logins. One week later five participants could not log into their accounts using passwords. Asterisks stand for a statistically significant difference ($p < 0.05$ in χ^2 test).

	On the first day	One week later
Password	*24.9 (12.4)	44.6 (30.7)
WebTicket	*35.8 (13.2)	30.5 (12.2)

Table 3. Login time in seconds. On the first day login using WebTicket took longer than that using passwords. However, one week later, there was no statistically significant difference between the two conditions. Asterisks stand for a statistically significant difference ($p < 0.05$ in Mann Whitney U test).

In the WebTicket condition, some participants had difficulty in finding which icon they had to click to launch the WebTicket reader dialog. We also observed that some participants opened the home pages and typed their email addresses before launching WebTicket's reader dialog. This process was unnecessary because WebTicket automatically complete the process when QR code was scanned. These observations indicated that people were not entirely clear about WebTicket's login process. Thus, we added the instruction on the reverse side of tickets after the first user study to clarify the login process (see Figure 1).

	Account Creation		Login	
	Ease	Length	Ease	Length
Password	2.1 (0.86)	2.5 (0.94)	2.4* (1.1)	2.3* (1.1)
WebTicket	3.1 (0.51)	2.9 (0.64)	3.1* (0.9)	3.2* (0.7)

Table 4. Participants' self-reported evaluations (means and standard deviations) about account creation processes and login processes (1 is very difficult or very long, 5 is very easy or very short). Asterisks stand for statistically significant differences between the password and the WebTicket conditions ($p<0.05$ in Mann-Whitney U test).

At the end of the second session, in a post-test survey, we asked participants whether the login processes using passwords and WebTicket were easy or difficult, and short or long using 5-point Likert scales (Table 4). In terms of ease, participants reported the login process using passwords as rather difficult (2.4), and using WebTicket as neither easy nor difficult (3.1). The difference between the results in the password condition and the WebTicket condition was significant ($p<0.05$ using Mann-Whitney U test). In terms of length, participants reported the login process using passwords as long (2.3), and using WebTicket as neither short nor long (3.2). The difference was significant ($p<0.01$ using Mann-Whitney U test). These results indicated that, although the actual login time using WebTicket was not shorter than the process using passwords, participants perceived that authentication using WebTicket was faster and easier than that using passwords.

USER STUDY #2: EVALUATING PHISHING RESISTANCE

We conducted the second user study to investigate phishing resilience of WebTicket as well as mobile phone version of WebTicket. The study had two sessions: an initial session and a follow-up session one week later. The initial session was similar to the first study, with an extra condition for mobile phone WebTicket. However, the second session was quite different. We asked participants to handle emails, including a phishing email, asking them to complete some tasks in a role-playing scenario.

Participants

We recruited 35 participants using an existing university recruitment web site. Twenty-nine participants completed the study. None of them participated in the first user study. Their ages ranged from 19 to 57 with a mean age of 30.1 (σ=11.1). Sixteen participants were males and 13 were female. They consisted of 19 university students, two university staff, and eight domestic residents. None of the students were computer science majors. We paid $10 USD for their participation and also paid additional $2 USD for each successful login in the follow-up session. In the initial session, we did not yet tell that they could get the bonus for successful logins. Thus, the bonus did not affect the participants' choice of passwords, while the bonus gave the participants an incentive to login in the follow-up session.

Procedure

In the initial session, we explained WebTicket to participants. Then, we asked the participants to create two accounts as a warm-up task, using both the paper-based and mobile phone versions of WebTicket. We then had participants create three accounts in three different mock e-commerce web sites, using password, paper-based WebTicket, and mobile-phone-based WebTicket in randomized orders. A change from the first study was that participants could choose *any* password in the password condition. Additionally, in the two WebTicket conditions, WebTicket wizard was *not* opened initially. Thus, the participants had to open it by themselves. We made these two changes to address the limitations in the first study.

The web sites were same in the first user study except that we added more fields in their registration forms, such as a credit card number field, to make them fit our phishing evaluation. We provided a persona of a university staff, whose identity the participants used to create these accounts. After creating the accounts, we asked the participants to log into their accounts one by one. We also increased the number of dummy tickets to 20 to further evaluate scalability of WebTicket. At the end of the initial session, we gave participants their tickets and asked them to bring them back to the follow-up session.

In the follow-up session, we evaluated the phishing resilience of the three authentication systems following the procedure used in Egelman et al.'s study [9] as much as possible. We asked participants to handle five emails from a professor. Three of them asked to purchase products using the accounts created in the first session. The other two emails were distracters. When purchasing products, the participants received three confirmation emails and one phishing email. The phishing email was same as one used in Egelman et al.'s study, although we modified shop names and URLs to make it fit with our study. The phishing email said that an order would be delayed and that, unless the participants clicked a link in the email to approve the delay, the order would be canceled. When the participants clicked the link, they went to a simulated phishing web site.

We divided participants into five conditions. Participants in the first condition received a phishing email against their password-based accounts. Participants in the next two conditions received a phishing email against their paper-based WebTicket accounts. Participants in the last two conditions received a phishing email against their mobile-phone-based WebTicket accounts.

We used two different kinds of phishing attacks on both paper-based and mobile-phone-based WebTicket. The first kind was a standard phishing attack that tried to trick people into using their username and password to log into a fake site. We call this *general phishing*. The second kind was a new attack that specifically targeted WebTicket itself, tricking people into giving up their QR codes by using a fake dialog (Figure 5). We define this phishing as *WT phishing*. In the current WebTicket, the QR code by itself does not allow attackers to access users' accounts because the information embedded in QR code is encrypted. However, we thought that the investigation of WT phishing could contribute to the evaluation of alternative designs (e.g., not using encryption).

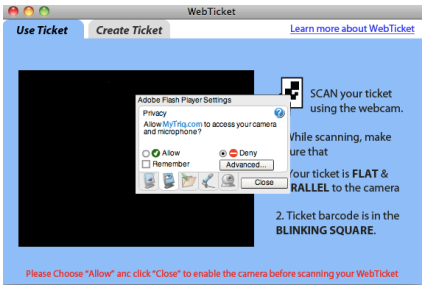

Figure 5. Fake reader dialog implemented by Flash. Users must click allow, then, close to enable a webcam. At the bottom, there was an instruction asking to do so.

In WT phishing, the fake dialog was implemented by Flash; hence it had to show the Flash privacy setting dialog asking for access to a webcam. If users choose *allow* and click *close*, the fake dialog worked in the same way as WebTicket. We decided not to show any border around the dialog because attackers would not know which border to use, and because users are not good at noticing that something is missing [7]. Moreover, we added an instruction asking users to choose *allow* and click *close* to *help* users to be phished (Figure 5).

RESULTS OF USER STUDY #2

Account Creation and Login

Table 5 shows account creation times and login times. As described in the previous section, participants had to launch the WebTicket wizard first in the WebTicket conditions. Additionally, in the password condition, we allowed participants to use any passwords, including reusing existing passwords. Although account creation time increased as a result of modification of registration forms, the results were in line with those of the first user study.

In the second session, we asked participants to handle five emails. Three of the emails asked the participants to log into their accounts that they created in the first session, and purchase products. Table 6 shows the success rates of the logins. Although we allowed participants to choose any passwords, there were still statistically significant differences between the success rates of passwords condition and that of the two WebTicket conditions.

	Account Creation	Login
Password	*113.8 (47.1)	27.0 (14.6)
Paper-based WebTicket	*192.3 (61.7)	30.3 (10.8)
Mobile-phone WebTicket	*165.1 (75.6)	32.5 (12.3)

Table 5. Account creation times and login times in seconds. The results complied with the results in the first user study. There were statistically significant differences in account creation times between the password and the two WebTicket conditions ($p < 0.05$, in Mann-Whitney U test).

	On the first day	One week later
Password	100% (29/29)	*90% (26/29)
Paper-based WebTicket	100% (29/29)	*100% (29/29)
Mobile-phone WebTicket	100% (29/29)	*100% (29/29)

Table 6. Success rates of the logins. Asterisks stand for statistically significant differences between the password and the two WebTicket conditions ($p < 0.05$ in χ^2 test).

Phishing Resilience

In the second session, we also investigated phishing resilience of passwords and WebTicket. We found that seven participants opened the phishing email and close it right away without reading its content possibly to save their time. Thus, we analyzed only the participants who actually read the phishing emails. Moreover, because there was no statistically significant difference between paper-based WebTicket and mobile-phone-based WebTicket, we combined these two conditions in the following analyses.

As shown in Table 7, all participants who read the general phishing emails in the password condition were phished. In contrast, using WebTicket, none of the participants were phished by general phishing emails. This result showed that WebTicket prevented general phishing attacks effectively, although it was not surprising. Because participants did not know their own passwords with WebTicket, they could not fall for a large class of phishing attacks.

In the WT phishing condition, seven participants out of ten did not scan their tickets. This is a significant improvement

compared to the password condition where all participants were phished ($p<0.05$ in Yates χ^2 test). According to our post-survey, five participants noticed the fake dialog's border did not match the border on their tickets. This illustrates that dynamic security skins helped our participants detect the fake dialog. Two other participants did not notice the difference, but clicked *close* on the Flash privacy setting dialog and found that the webcam was not working. They closed the fake dialog and opened the legitimate reader dialog to scan their tickets.

	General Phishing	WT Phishing
Password	100%* (6/6)	-
WebTicket	0%* (0/6)	30%* (3/10)

Table 7. The ratios of phished participants out of those who read the phishing emails. The differences between general phishing against password and other two conditions were statistically significant ($p < 0.05$ in Yates' χ^2 test). Note that for WT Phishing, attackers still need to obtain a key stored in users' computers to decrypt the barcode.

Another three participants scanned their tickets using the fake dialog, meaning that attackers would have obtained the barcodes on their tickets. However, the attackers would still need to obtain the key stored in their computers to access the accounts, unlike standard phishing attacks that would allow immediate access.

These results indicate the importance of defense in depth. For the seven participants, the dynamic security skin or the privacy dialog prevented the attack. Besides, while three participants scanned their tickets, WebTicket would have prevented access to their accounts because of its encryption.

USER STUDY #3 FIELD TRIAL

To evaluate how well WebTicket works in practice, we conducted a field study looking at WebTicket's applicability in real world environments. We asked participants to use WebTicket on their personal computers using their actual accounts for three weeks.

Participants

We recruited participants who used computers running MacOS with a webcam, used Firefox as a primary web browser and had access to a printer. We recruited 10 participants using an existing university recruitment web site. None of them participated the previous studies. Their age ranged from 19 to 42 with the median age of 32. Three participants were male and seven were female. They consisted of five university students, four employed and one unemployed. We paid $35 USD for their participation.

Procedure

On the first day, we asked participants to install WebTicket. We also asked them to create three paper-based tickets for three existing accounts that they accessed once a day, once a week, and once a month. If they did not have accounts that exactly matched the criteria, we asked them to choose the closest ones. Although we did not ask the participants to change their passwords, we asked them to use their tickets whenever they logged into the accounts. Furthermore, we recorded how (e.g., typing passwords or using tickets) and when they logged into their accounts using a logging application. The ticket creation step was conducted in our lab for five participants using their laptops, and in their offices or homes for the other participants using their desktop computers. One participant had an Android phone and created mobile-phone-based tickets also. Moreover, two participants with multiple computers setup WebTicket on all their computers and shared the encryption key among the computers. Three weeks later, we asked participants to complete a survey and conducted interviews.

RESULT OF USER STUDY #3

Participants created tickets for a wide variety of accounts including, SNS (e.g., Facebook), email (e.g., Gmail), ecommerce (e.g., Amazon.com), and finance (e.g., Chase). After they created tickets, we asked how easy or difficult it was to install WebTicket and to create tickets for their accounts using 5-point Likert scale. Our participants rated installation as very easy (mean of 4.6 and median of 5) and ticket creation as easy (mean of 4.2 and median of 4).

According to our data logs, our participants accessed their accounts using WebTicket with a range of 21 to 150 times during the study period with a median of 52 times. They did not access their accounts by typing passwords as we requested. Furthermore, in our post-experiment survey, our participants reported that they did not access these accounts from other computers during the study period. We also asked participants whether they wanted to use the WebTicket for some of their accounts based on their experiences using 5-point Likert scale (1 is strongly disagree and 5 is strongly agree). Their answers ranged from 3 to 5 with a mean of 4 and median of 4.

In the interview, P3 commented, "WebTicket is very easy, functional, and safe. From what I understand, the URL, username, and password are all encrypted into the ticket and can only be unencrypted by your machine which recorded the actions and printed the ticket". This indicated that P3 understood how the WebTicket worked and appreciated both its usability and security.

On the other hand, P5 and P6 showed concerns about carrying paper-based tickets. P5 said, "[It is] kind of annoying to have the papers constantly available for when you want to log on". P5 also said, "[the mobile-phone-based ticket] would be much better and efficient because I am more likely to have my phone at hand and would prefer to use it instead of saving the little papers". This implies that, although the portability issue could be a limitation for some users, the mobile-phone-based ticket could mitigate it.

In contrast, P7 mentioned an advantage of having paper-based tickets. He said, "I'm concerned [about] using a too-easy password for my bank account. But, if I use a complicated password I have to write down the password, or save it in my laptop. I don't want to do so. I'm carrying around my laptop. But, [using paper-based ticket], I can keep the password at home with some protection."

P9 with mobile-phone-based ticket also commented, "I prefer the cellphone version because I don't have to carry pieces of paper. [...] I'm a little bit concerned about losing my phone. I may use the paper one for really important accounts, such as my bank account. I don't use it that frequently. So, I can do it at home." These comments suggest that paper-based tickets and mobile-phone-based tickets could be used for different types of accounts.

Besides, P2 said, "I like WebTicket does not store a copy of my password on some server but rather is kept locally on my machine or ticket." This implies that he appreciated the fact that WebTicket did not rely on third parties to be used.

Another interesting observation was that no participant mentioned being able to use ticket only their computers as a drawback when asked the drawbacks of WebTicket, while three participants regarded it as a benefit in terms of security. This could be because people mostly use their own computers to access accounts [15]. Although the relatively short study period could bias users' perceptions, this observation implies that not allowing people to use tickets on foreign computers is less likely to be a major limitation.

DISCUSSION

Our results indicate that participants were not slower than passwords when logging in with WebTicket. Participants also reported the login process with WebTicket as easier and shorter than with passwords. These results are very encouraging considering that the strong passwords and the phishing resistance provided by WebTicket.

Note that we do not intend to replace *all* passwords with WebTicket. Instead, we want users to choose authentication schemes according to their needs. For frequently used accounts, passwords may make more sense because these passwords are less likely to be forgotten and fast to enter in. In contrast, for occasionally used accounts, WebTicket will make more sense because WebTicket can provide a reliable way to login. Additionally, because WebTicket does not require server side modifications, users who have difficulty in memorizing passwords, such as memory impaired or novice users, could choose WebTicket for a site, while others use passwords for the same site.

Portability could be a challenge with paper-based tickets. For example, in our field study, two participants showed concern about carrying many tickets. However, other participants commented on the *benefit* of keeping paper-based tickets at home. Considering that people access their accounts mostly from work or home [15], there would be potential benefit in keeping tickets at these locations for many people. For exceptions where users need repeated accesses to these accounts in other places, they could use mobile-phone based tickets or simply use passwords.

ALTERNATIVE DESIGNS FOR WEBTICKET

We investigated one possible point in the design space for WebTicket. However, there are cases where different design choices may make more sense. One major design decision was whether to have one ticket for each account, or just have one ticket for multiple accounts. If we had opted for the latter, the passwords would need to be stored on the computer itself and the ticket would simply be the key to access those passwords. The advantage is that users do not need to carry many tickets. The disadvantages are in synchronizing account information when new accounts are created (if there are multiple computers).

We also considered a hybrid approach, having one ticket per important account, and one ticket for all "unimportant" accounts. In this case, the ticket for unimportant accounts works in a similar way as PwdHash [26]. In this design, synchronization does not matter because passwords are generated dynamically. However, this design has challenges dealing with site-specific password composition policies.

Another important design choice is whether we should encrypt the information embedded on a ticket. Without encryption, users can log into their accounts from any computer as long as WebTicket is installed. However, past work has found that users mostly access their accounts from their own computers at work or home [15]. As such, we felt that the benefit of allowing access on foreign computers was small considering that allowing it would expose users to a large number of vulnerabilities, including theft of tickets as well as cameras taking pictures of the QR codes.

LIMITATIONS

One constraint is that WebTicket requires webcams, and either printers or smartphones. Given that these devices are commodities, we feel that this is an acceptable tradeoff.

Scalability is a major limitation of paper-based tickets. However, we do not intend to replace *all* the passwords with WebTicket. For people who have many accounts, the mobile-phone version can be used to manage tickets. Durability is another limitation of paper-based tickets. For example, if a ticket is crumpled or washed, the QR code on the ticket cannot be scanned. Then, users have to reset their passwords in corresponding websites and renew tickets.

A potential weakness with paper-based tickets is that people might leave them near their computers, in which case these tickets are only slightly better than a post-it note with passwords on them (since people can't use the ticket on other computers). However, depending on one's threat model, this may not be a huge security risk, especially if one has good physical security or login passwords that restrict access to computers.

There could be some cases where users want to access their accounts from foreign computers, although people use their own computers most of the times [15]. There are possible workarounds, e.g. storing the master key on users' mobile phones. However, these approaches increase the potential for loss if the mobile phone is lost or stolen.

CONCLUSIONS

In this paper, we introduced WebTicket, a novel web account management system that transforms knowledge-based authentication into token-based authentication using

commodity devices such as webcams, printers, and mobile phone without the need to modify server side systems.

Through two lab studies, we found that participants reported that the account creation process using WebTicket was not worse than the account creation process using password, and that the login process using WebTicket was easier and shorter than the login process using passwords. WebTicket also provided reliable authentication after one week and good phishing protection. Furthermore, after the three weeks of a field study, the participants were positive about WebTicket. Although further investigations of its limitations may be necessary, we believe that WebTicket works quite well among good portion of accounts and users.

REFERENCES

1. eToken. http://www.aladdin.com/etoken/.
2. QRCode.com. http://www.denso-wave.com/qrcode/.
3. RSA securID http://www.rsa.com/node.aspx?id=1156.
4. A. Adams and M. Sasse. Users are not the enemy. Communications of the ACM (1999).
5. S. Brostoff and M. Sasse. Are passfaces more usable than passwords: A field trial investigation. In Proc. of HCI 2000, (2000).
6. S. Chiasson, R. Biddle, and P. V. Oorschot. A second look at the usability of click-based graphical passwords. In Proc. of SOUPS (2007).
7. R. Dhamija and J. Tygar. The battle against phishing: Dynamic security skins. In Proc. of SOUPS (2005).
8. A. Dirik, N. Memon, and J. C. Birget. Modeling user choice in the passpoints graphical password scheme. In Proc. of SOUPS (2007).
9. S. Egelman, L. F. Cranor, J. Hong. You've been warned: an empirical study of the effectiveness of web browser phishing warnings. In Proc. of SIGCHI (2008)
10. K. Everitt, T. Bragin, J. Fogarty, and T. Kohno. A comprehensive study of frequency, interference, and training of multiple graphical passwords. In Proc. of CHI (2009).
11. Gartner. Automated password resets can cut it service desk costs. 2004.
12. S. Gaw and E. Felten. Password management strate-gies for online accounts. In Proc. of SOUPS (2006).
13. J. T. Hallinan. Why We Make Mistakes. Broadway, 2009.
14. E. Hayashi, R. Dhamija, N. Christin, and A. Perrig. Use your illusion: secure authentication usable anywhere. In Proc. of SOUPS (2008).
15. E. Hayashi, J. I. Hong, A Diary Study of Password Usage in Daily Life. In Proc. of SIGCHI (2011).
16. D. V. Klein. ”foiling the cracker”: A survey of, and improvements to, password security. In Proc. of USENIX Security, (1990).
17. S. Klemmer, M. Newman, and R. Farrell. The designers’ outpost: a tangible interface for collaborative web site. In Proc. of UIST (2001).
18. C. Kuo, S. Romanosky, and L. Cranor. Human selection of mnemonic phrase-based passwords. In Proc. of SOUPS (2006).
19. S. L. Learning 10,000 pictures. Quarterly Journal of Experimental Psychology, (1967).
20. W. MacKay. Is paper safer? The role of paper flight strips in air traffic control. ACM Transactions on Computer-Human Interaction, (1999).
21. J. McCune, A. Perrig, and M. Reiter. Seeing-is-believing: Using camera phones for human-verifiable authentication. In IEEE S&P (2005).
22. D. McGee, P. Cohen, R. Wesson, and S. Horman. Comparing paper and tangible, multimodal tools. In Proc. of CHI (2002).
23. T. Moran, E. Saund, W. V. Melle, A. Gujar, K. Fishkin, and B. Harrison. Design and technology for collaborage: collaborative collages of information on physical walls. In Proc. of UIST (1999).
24. L. Nelson, S. Ichimura, E. Pedersen, and L. Adams. Palette: a paper interface for giving presentations. In Proc. of CHI (1999).
25. A. Paivio and T. Rogers. Why are pictures easier to recall than words? Psychonomic Science, (1968).
26. B. Parno, C. Cuo and A. Perrig, PhoolprofPhishing Prevention. In Proc of. the Financial Cryptography and data security (2006).
27. B. Ross, C. Jackson, N. Miyake, D. Boneh, and J. Mitchell. Stronger password authentication using browser extensions. In Proc. of the USENIX Security(2005).
28. M. Sasse, S. Brostoff, and D. Weirich. Transforming the ‘weakest link’— a human/computer interaction approach to usable and effective security. BT technology journal, (2001).
29. A. Whitten and J. Tygar. Why johnny can’t encrypt. In USENIX Security, (1999).
30. S. Wiedenbeck, J. Waters, J. Birget, and A. Brodskiy. Passpoints: Design and longitudinal evaluation of a graphical password system. International Journal of Human-Computer Studies, (2005).
31. J. Yan, A. Blackwell, R. Anderson, and A. Grant. Password memorability and security: Empirical results. In IEEE Security & privacy, Vol. 2, pp. 25–31, (2004).
32. K. Yee, K. Sitaker. Passpet: convenient password management and phishing protection. In Proc. of SOUPS (2006).
33. Your Top 20 most frequently used passwords. http://www.tomshardware.com/news/imperva-rockyou-most-common-passwords,9486.html