A Billion Keys, but Few Locks: The Crisis of Web Single Sign-On

San-Tsai Sun, Yazan Boshmaf, Kirstie Hawkey, Konstantin Beznosov
Laboratory for Education and Research in Secure Systems Engineering (LERSSE)
Department of Electrical and Computer Engineering
University of British Columbia
Vancouver, Canada
santsais,boshmaf,hawkey,beznosov@ece.ubc.ca

ABSTRACT

OpenID and InfoCard are two mainstream Web single sign-on (SSO) solutions intended for Internet-scale adoption. While they are technically sound, the business model of these solutions does not provide content-hosting and service providers (CSPs) with sufficient incentives to become relying parties (RPs). In addition, the pressure from users and identity providers (IdPs) is not strong enough to drive CSPs toward adopting Web SSO. As a result, there are currently over one billion OpenID-enabled user accounts provided by major CSPs, but only a few relying parties.

In this paper, we discuss the problem of Web SSO adoption for RPs and argue that solutions in this space must offer RPs sufficient business incentives and trustworthy identity services in order to succeed. We suggest future Web SSO development should investigate and fulfill RPs' business needs, identify IdP business models, and build trust frameworks. Moreover, we propose that Web SSO technology should build identity support into browsers in order to facilitate RPs' adoption.

Categories and Subject Descriptors

D.4.6 [**Security and Protection**]: Authentication

General Terms

Security, Privacy

Keywords

Web Single Sign-On, Web Identity Management, Authentication, OpenID, InfoCard

NSPW'10, September 21–23, 2010, Concord, Massachusetts, USA.

1. INTRODUCTION

With Web 2.0, the user is both a consumer and provider of Web content. However, today's Web is site-centric. A user has to maintain a separate copy of their identity and corresponding password for each content-hosting and service provider (CSP). The site-centric Web is the root cause of many problems current web users are facing. A 2007 large-scale study of password habits found that a typical web user has about twenty-five accounts that require passwords, and types eight passwords per day [19]. Web users face the burden of managing this increasing number of accounts and passwords, which leads to "password fatigue" [71]. Aside from the burden on human memory, password fatigue may cause users to devise password management strategies that degrade the security of their protected information [22, 19]. In addition, the site-centric Web makes online profile management and controlled personal content sharing difficult as users' profiles and access control policies are restricted to a single administrative domain [63].

Web single sign-on (SSO) systems are built to address the root causes of the site-centric Web problem. A Web SSO system separates the role of identity provider (IdP) from that of relying party (RP) to enable users to leverage one identity across multiple RPs. An IdP issues identities or credentials to users, while an RP depends on the IdP(s) to assert the user credentials before allowing access to its services.

OpenID [57] and InfoCard [47] are mainstream Web SSO solutions targeted for Internet-scale adoptions [38, 13]; however, they are facing RPs' adoption problem. Evidence shows that although major CSPs acted quickly to become IdPs, only a limited number of websites have adopted the role of RPs [54, 33, 46, 53]. This is similar to having more than a billion keys, but few locks to use. Figure 1, the broken triangle of the Web SSO identity ecosystem, illustrates this RP adoption problem. Each dashed line indicates a lack of driving force or incentive between two actors in the ecosystem. For instance, line `B` shows the lack of incentive from a relying party for supporting SSO for users.

The main reason behind this RP adoption problem is that the business model of current Web SSO systems does not provide CSPs with sufficient motivation to become RPs. While there are strategic benefits (e.g., marketing, thought-leadership reputation) for being an IdP, there are business risks and no immediate benefits when becoming an RP.

Fundamentally, Web SSO systems shift the functions of

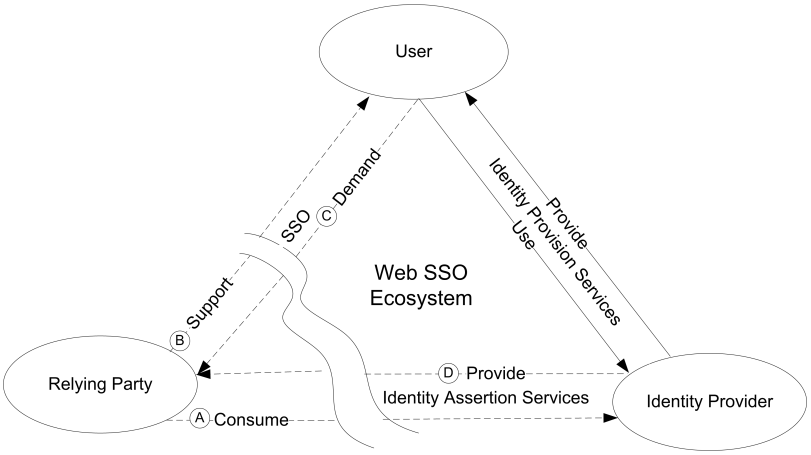

Figure 1: The broken triangle of the Web SSO identity ecosystem. Each dashed line indicates a lack of driving force or incentive between two actors.

identity collection and authentication from RPs to IdPs. However, the incentive for RPs to rely on the identity assertion services provided by IdPs is insufficient, as illustrated by line `A` in Figure 1. RPs are not willing to relinquish control over their user base unless they can obtain user data and verify the IdP's authentication and data collection policies [35, 13]. In addition, RPs have to choose which IdPs to trust as they are liable for the loss when IdPs get compromised [45].

From a user-service provision point of view, the incentive for an RP to support Web SSO is lacking as well (line `B` in Figure 1). CSPs are reluctant to modify their login UI and process because new login procedures might confuse and upset users [20, 58]. In addition, RPs might not want to expose their users to potential business competitors because once the attention of users has been redirected to an IdP during the login process, they might not return [13]. Furthermore, adopting SSO does not provide RPs with a competitive advantage, as users are unlikely to choose one RP over another simply because of Web SSO. As early adoption does not appear to provide RPs with competitive advantages, CSPs may be waiting until Web SSO technology is mature and the cost of user training has already been absorbed by other websites.

To encourage adoption by RPs, Web SSO systems have to rely on demand from users as the driving force (line `C` in Figure 1). However, users have no urgent need for SSO as they could use a password manager as a limited version of a personal identity manager [42]. Additionally, the user experience provided by today's Web SSO solutions is inconsistent and counter-intuitive, which imposes a significant cognitive burden on average web users [20, 58, 13]. Moreover, users might not want to risk their security (e.g., single-point of failure [38], phishing attacks [36, 13, 40]) and privacy (e.g., IdP tracking [38], unintentional disclosure [23, 72, 25]) for SSO.

Another driving force for improving the RP adoption rate could come from new identity validation services (e.g., age over 18, clean criminal record) provided by IdPs (line `D` in Figure 1). RPs could utilize those services to reduce operational costs or create novel business opportunities. However, based on the experiences of public certification authorities, asserting identity has yet to become a profitable business [8]. As the cartoon "on the Internet, nobody knows you're a dog" [61] illustrates, verifying a user's true identity is inherently difficult on the Web. In addition, because the Web is a virtual world beyond the boundaries of physical governments, even if some organizations would like to provide user attributes certification services (e.g., date of birth), those solutions would be domain-specific and may not scale well (e.g., RPs might not trust such an organization).

The broken Web SSO triangle is the result of a combination of usability, security, privacy, trust, business, and legal problems. In this paper, we use the metaphor of the broken Web SSO triangle to identify the underlying forces behind the RP adoption problem. We argue that future revisions to the Web SSO technology must provide RPs with concrete business benefits and trustworthy identity services in order to achieve widespread adoption. We suggest future Web SSO development should investigate and fulfill RPs' business needs, identify IdP business models, and build trust frameworks for IdPs to provide trust-worthy identity-attribute assertion services. In addition, we suggest future Web SSO technology should build identity support directly into the browser in order to address usability, security, and privacy problems and to function as a platform to facilitate RPs' adoption.

The rest of the paper is organized as follows. The next section presents background and related work. Section 3 expands the discussion of the Web SSO broken triangle problem further. Section 4 presents our recommendations for the future Web SSO development, and Section 5 summarizes the paper.

2. BACKGROUND AND RELATED WORK

In this section, we provide background on password managers and Web SSO systems, and present related work on browser-supported Web SSO systems. Readers familiar with the subject can proceed directly to the next section.

2.1 Password and form managers

One solution to reduce the burden on memory and the overhead of credential management are password managers, which help web users organize their online user names and passwords [44]. A recent study [22] found that the most

commonly used are those built-in to the browser itself (e.g., password auto complete), rather than those implemented as a browser extension (e.g., Password Multiplier [26]). Password managers typically store encrypted password data in a local database or file and are able to automatically fill in login forms of registered web sites.

Password managers can reduce a user's memory burden as they only need to remember a single master password [44]. However, users may have difficulty migrating their existing passwords to the system [10]. Such systems typically have issues with transportability of passwords between computers [10, 44], and users may not trust the security of these systems [22]. In addition, for password managers that improve security through custom generated passwords (e.g., Passpet [78]), users may be uncomfortable not knowing the actual site passwords [10].

Sxipper [65] is a form manager that helps users fill-in web forms during registration or ordering processes. It also recognizes login forms and leverages the Firefox built-in password manager for single-click login. This Firefox add-on enables web users to maintain several personae; each persona consists of a set of user attributes such as name, email, and address. When encountering a web form, Sxipper prompts the user to pick a persona and then fills the corresponding form automatically. The main limitation of Sxipper is that it might not detect forms correctly. In addition, Sxipper stores sensitive information such as credit card numbers on the user's local machine. This poses a security threat if the user's computer is compromised, and raises portability issues when users switch between computers or want to use a shared or public computer.

Password and form managers can be used by web users to reduce the friction of the sign up and sign on process. However, unlike Web SSO solutions, users still have to maintain multiple unnecessary accounts, which makes online profile management and controlled personal content sharing difficult.

2.2 Single and multiple-domain Web SSOs

To achieve Web SSO, major CSPs have provided a way for other CSPs to accept user credentials from their domain (e.g., Microsoft Live ID [56], Yahoo BBAuth [76], AOL OpenAuth [4]). However, these systems are proprietary and centralized; identity information is maintained and controlled by a single administrative domain. Similarly, intra-organization Web SSO systems provide the ability for web-based applications across an organization to rely on a shared user database, and to provide user access to applications with minimal number of sign-ons. Examples of such systems include PubCookie [69], CAS [77] (Yale University), and CoSign [68] (The University of Michigan). These systems are generally open source and HTTP cookie-based solutions. The main limitation of both types of system is their closed nature.

Federated identity solutions enable cross-domain single sign-on, and remove the need for users to keep identifiers and passwords at individual CSPs. In a federated domain, IdPs supply assertions about a user's identity, while RPs "consume" provided identity and mediate accesses based on this information. Solutions such as coalition-based access control (CBAC) [11], Liberty Alliance Project [37], and Shibboleth [32] (based on SAML [50]) are examples of federated identity systems. However, these solutions require pre-established trust relationships and agreements between organizations in the federation, making them hard to scale on the Web.

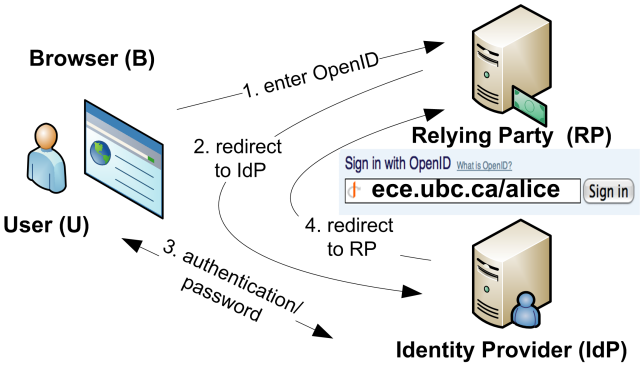

Figure 2: How OpenID works.

2.3 OpenID

OpenID [57] is an open and user-centric protocol for Web SSO. OpenID is user-centric in the sense that users are free to choose or setup their own OpenID providers. One key scalability feature of OpenID is that it does not require any pre-established trust relationships between IdPs and CSPs. According to the OpenID Foundation [53], as of September 2009, more than one billion OpenID enabled user accounts were provided by major CSPs (e.g., Google, Yahoo, AOL, Facebook).

OpenID [57] is an open and user-centric protocol for Web SSO. In OpenID, a user's identity is a URL, and the OpenID authentication process asserts that the user controls the content at that URL. The following steps demonstrate how the OpenID protocol works:

1. User **U** enters her OpenID i (e.g., ece.ubc.ca/alice) via a login form presented by a relying party **RP**, as illustrated in Figure 2.
2. **RP** makes an HTTP request on i to fetch a document (either an XRDS or an HTML document) that contains the OpenID IdP endpoint **IdP** (e.g., https://ece.ubc.ca/openid). **RP** then redirects **U** to **IdP**.
3. **U** authenticates with **IdP** (e.g., by entering her user name and password).
4. **IdP** verifies the credential and redirects **U** back to **RP** with an authentication token that **RP** can verify.

OpenID is a promising solution; however, there are three main issues that must be addressed in order to achieve global adoptions. First, OpenID focuses primarily on user authentication; but, phishing attacks have demonstrated that it is also important for the user to be able to assert the authenticity of IdPs [36, 40]. Phishing works by tricking a user into visiting a malicious replica of a web site. OpenID compounds this problem by "conditioning" users to use login forms to which they have been redirected by untrusted sites. Second, the OpenID identifier scheme has usability issues. OpenID uses a URL as an end-user's identifier; this acts as a universal user account and is valid across all CSPs. The main advantage of using a URL as an identifier is that it is tangible, clickable, and can function as a user's personal information portal to associate profile information and related services. However, web users perceive a URL as a

"web address" instead of a personal identifier; and it is difficult for average web users to understand and remember their OpenID [15, 3]. Finally, the OpenID protocol—as defined by its specification [57]—enables IdPs to track all websites a user has logged into with her OpenID account. The tracking capability of OpenID makes cross-site profiling easy and possible.

2.4 InfoCard

Information cards (known as InfoCard) [47] are personal digital identities that are analogous to real-world identity cards such as passports, driver licenses, and credit cards. Each card contains assertions about a user's identity that are either self-issued or issued by an identity provider. When logging into a web site, the user selects a card instead of typing a user name and password. Cards are managed on client computers by a software component called an identity selector (e.g., Windows CardSpace [41], Higgins Card Selector [66]). In June 2008, the Information Card Foundation [67] formed to advance the use of the InfoCard metaphor as a key component of user-centric identity systems. Industry leaders such as Equifax, Google, Microsoft, Novell, Oracle, PayPal, and VeriSign are among the steering members of the Information Card Foundation.

In order to use InfoCard the user must first create a self-issued card on her own machine using the identity selector or obtain an IdP-issued card from an identity provider. A newly issued InfoCard is an XML document that can be transmitted to the user via e-mail or Web download. The following steps and Figure 3 illustrate how a user uses InfoCard to log into a RP website:

1. User **U** uses browser **B** to make an HTTP request to a protected resource on an InfoCard-enabled relying party **RP**.
2. **RP** returns a security policy to **B** indicating what type of identity and channel security the service requires.
3. **B** invokes the identity selector **S** and passes in the security policy retrieved from **RP**. **S** shows the user a collection of cards that matches the given policy. The match is primarily determined by the type of token and claims required by the service.
4. Once the user selects a card to send, **S** initiates a security policy exchange conversation with the identity provider **IdP** that issued the card.
5. **IdP** returns a security policy to **S** indicating how the user is supposed to prove her credential.
6. **U** provides her credential to **S** (e.g., by entering user name and password).
7. **S** makes a request to **IdP** for the required claims along with **U**'s credentials.
8. **IdP** returns a security token to **S** for the user to send to the service. This token contains all the claims **RP** requested.
9. Based on the user's consent, **S** passes the security token to **B**.
10. **B** in turn, passes the security token to **RP**. **RP** then makes access decisions based on the received security token.

Information cards have important features such as phishing resistant authentication, IdP-to-RP unlinkability, and real-time user consent. However, in comparison to OpenID, InfoCard is a heavy-weight protocol. In particular, users need to install an identity selector and relying parties must have a valid SSL certificate configured to provide secure channels when communicating with identity selectors. Furthermore, as cards are stored on the local repository, they could be stolen and used to impersonate the victim if the user's machine is compromised [28]. Additionally, InfoCard raises privacy issues when it is used on shared or public computers, and it is difficult to use them if users switch between multiple computers.

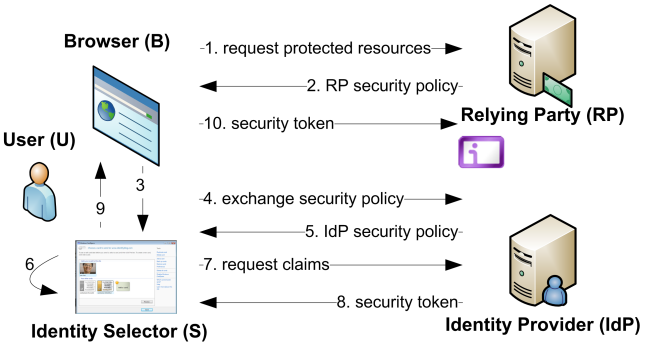

Figure 3: How InfoCard works.

2.5 Browser-supported single sign-on systems

VeriSign's Seatbelt Firefox add-on [70] is designed to make OpenID more convenient to use by automatically filling in a user's OpenID URL when visiting relying parties. This extension also provides an OpenID user with information about their login state with their OpenID provider and automatically monitors OpenID transactions to help prevent phishing attacks. Seatbelt is easy to use; however, it may not detect OpenID login form fields precisely as a simple text matching technique (e.g., openid, oidurl, open-id, open_id) is used to identify them. In addition, it does not work with RPs that use a list of IdP icons for user to choose from. Moreover, it requires Seatbelt specific configurations and login state provision from the participating OpenID IdPs. Furthermore, it is unable to detect "rogue relying party proxying" phishing attacks (e.g., content scraped from the real site) that do not rely on HTTP redirections when spoofing victims (see http://idtheft.fun.de/ for an example).

Weave Identity [43] from Mozilla Labs is a Firefox add-on that leverages a Firefox built-in password manager for single-click and automatic logins and integrates Weave server accounts for automatic OpenID sign-on. Similar to VeriSign Seatbelt, it might not detect and submit login forms correctly; and automatic OpenID login support is limited only to Weave accounts.

3. THE BROKEN WEB SSO TRIANGLE

Currently, there are many high-profile OpenID IdPs, but not enough CSPs that support OpenID as RPs. On the IdP side, there are over one billion OpenID-enabled user accounts provided by major CSPs such as Google, Yahoo, and AOL. However, there are relatively few RPs. A recent search found only 882 RPs on OpenID Directory [54], 240 RPs on MyOpenID Directory [46], and 40,000 found by the JanRain's relying party statistic project [33]. Compared to the total number of websites (213,000,00) available today [48], the adoption rate is less than 0.02%. In addition, a large portion of the adoption rate is contributed by some popular open source projects, such as WordPress, that support OpenID directly in the software. Furthermore, the major-

ity of high-profile OpenID IdPs such as Google, Yahoo, and Microsoft only act as an IdP, they do not allow users with OpenIDs from other providers to access their services. For InfoCard, the list of RPs and IdPs is almost empty [67].

Becoming an OpenID provider comes with strategic advantages for CSPs and the process is easy. All that is needed is to setup a new OpenID endpoint service to interface with their existing user accounts. As early adoption could change their online-identity market shares, major CSPs moved quickly to gain the marketing and thought-leadership benefits by simply being an IdP. On the other hand, being a RP can impose risks on their business with no direct returns. In the following sections, we expand our discussion for each broken link in the Web SSO triangle illustrated in Figure 1.

3.1 Lack of incentives for RPs to rely on IdPs

Web SSO systems shift the function of identity management from RPs to IdPs. However, as Murdoch and Anderson observe, few organizations are willing to trust a third-party organization to authenticate users when they have no recourse in the event of error or attack [45]. In the case of Web SSO, it is unlikely that CSPs will act as RPs if IdPs guard the identities while RPs pay the cost of failure. In addition, RPs are unlikely to be willing to use their potential competitors as IdPs [13]. CSPs will resist an architecture that lets a third party IdP manage and authenticate their users unless there is a very close relationship between them [35, 13].

The "identity war" has been ongoing since the beginning of the Web, and "walls" have been built by CSPs to protect their subscriber base [6, 63]. Even before OpenID and InfoCard, major CSPs have provided a way (e.g., Microsoft Live ID, Yahoo BBAuth [76], AOL OpenAuth [4], Google AuthSub [24]) for other CSPs to accept user credentials from their domain. However, most CSPs would like to maintain as many registered users as possible, and are reluctant to tear down their guarded walls [63].

3.2 Lack of incentives for RPs to support SSO

From the RP business point of view, migrating to Web SSO is not a worthwhile endeavor. First, the advantages of being an RP are operational rather than competitive. Web SSO could help CSPs reduce their password maintenance and recovery costs and allow them to access user profile data provided by IdPs during a registration process. However, those benefits would not give RPs competitive advantages as web users are unlikely to choose one RP over another solely because the RP has supported SSO for login.

Second, confusing user experiences could upset users; and, as a result, impact a CSP's business directly [13]. Thus, CSPs might not want to simply add an OpenID input field or InfoCard icon on their existing login forms because doing so might confuse users. Yahoo conducted an OpenID usability study with experienced Yahoo users to understand their mental models and the usability issues associated with enabling Yahoo OpenID on an RP website [20]. The study found that most participants were confused by login screens which contained both the traditional username/password login form and the OpenID URL textbox. Participants were also distracted by additional steps (e.g., setup of custom identifier, CAPTCHA page, "Let Me In" page) on Yahoo website. Once redirected to Yahoo, some participants were not able to find the "right door" to return back to the RP website.

Third, even with improved login interfaces (e.g., JanRain RPX [31]), most RPs still want their users to create an account on their sites for the purpose of monitoring usage, preventing abuse of their service, and protecting information about their customers [13]. For first-time users of an RP website, combining single sign-on with additional registration steps could confuse them even further [20, 58]. New user sign-up is a critical process for business; most CSPs would not want to risk their potential customers for the sake of SSO.

Technically, it is not difficult to support Web SSO as an RP [53, 67]. However, as there are no direct benefits can be expected, CSPs would rather wait until Web SSO technology is mature and pervasive. CSPs will likely embrace Web SSO only when the expected returns are much higher than the risks.

3.3 Lack of driving forces from users

As the business model of Web SSO solutions does not provide CSPs with sufficient motivation to become RPs, those solutions have to rely on user demand to drive adoption by RPs. However, the driving forces from users are not significant enough to overcome the resistance of CSPs for being RPs as we explain in the following subsections.

3.3.1 No urgent needs for Web SSO

Without SSO, web users tend to use weak passwords or use the same passwords across CSPs as choosing strong memorable passwords is a challenging task [2]. Nevertheless, as Florencio et al. [19] found, strong web passwords accomplish very little for websites that employ a lockout mechanism. When a lockout mechanism can restrict brute-force attacks, a simple 6-digit password would be sufficient. As there is no direct data and user experiences available indicating weak-password leads to physical asset loss, most users are "conformable" with weak-password practices [29].

For those websites that require strong passwords, users often choose password managers to reduce their memory burdens [22]. Many web users use the password manager feature in the browser to turn their browser into a limited version of identity manager [42]. Password managers are inconvenient when users switch between computers or when they want to use shared or public computers; however, those are occasional events that are considered tolerable by most users [22].

3.3.2 Inconsistent and counter-intuitive user experience

The interaction flows in current Web SSO solutions follow a "shared-identity" sign-on (SISO) paradigm rather than a true "single" sign-on paradigm. With SISO solutions, users can use one identity to sign into multiple RPs. Nevertheless, when accessing N RPs using one IdP, the user must visit $N+1$ different login forms (one for each RP website and one on the IdP), choose an IdP to login N times via N possible ways, read the consent page on the IdP N times, and log out $N+1$ times through $N+1$ different interfaces. Figure 4 shows some screen shots of current RP login forms. These inconsistent and counter-intuitive user experiences impose significant cognitive burden on average web users [20, 58, 13].

Figure 4: Inconsistent RP login forms.

SISO redirects users' attention during the login process. However, studies have shown that once the attention of users has been redirected, they might not return [20, 13]. Users may be distracted by the advertisements posted on the IdPs or derailed by navigating through the details of informational links. Depending on each RP's implementation, users who manage to return from the IdP login/consent page might be landing on a different page other than the one they were attempting to access (e.g., shopping cart → login form on RP → login form on IdP → consent form on IdP → user account page on RP). Users can be confused when they are not returned to the task at hand after a successful authentication [20]. In addition, redirection exposes users to phishing attacks and makes user tracking by the IdP possible. InfoCard redirects users only to their identity selector. However, for N RPs using one IdP, the user has to provide her credential to the IdP N times unless a self-issued card (without password protection) is selected. Using multiple identities in one browsing session confuses users further. When users sign onto multiple IdPs in one browser session, they have to remember which identities were used for accessing which RPs. Mixing identities in one browser session makes it difficult for users to determine why an access failed or who to contact when a problem is encountered.

SISO requires usable interfaces and login flows from both RPs and IdPs. Simply adding an OpenID textbox or InfoCard icon to the traditional login page is not an option [20, 58]. To improve the user experience, some OpenID RP adopters provide a list of IdP logos on their login form for users to choose from, as illustrated in Figure 4. The users can simply click on an IdP icon to initiate a sign-on process. However, this approach leads to the "NASCAR" problem [40] when the list of IdPs grows too long to fit on the login screen. In addition, using a list of IdPs restricts the users' freedom of choice, which impairs healthy competition in the ecosystem.

SISO is especially problematic in Web 2.0 applications that require access to personal data located on multiple CSPs. For OAuth-based applications [51] that process a user's personal content from different providers, being presented with a login form on each CSP is annoying, and imposes a cognitive burden on the user [5]. For client-side mashups that use Ajax-style web services to acquire user data from several websites, login forms would block such communications. In addition, SISO-based solutions may be more difficult to use on mobile devices that have limited input capabilities.

3.3.3 Security and privacy concerns

In addition to the usability issues, security and privacy concerns are other factors that hinder user demand for SSO. One inherent risk of using Web SSO is that one compromised account on an IdP can result in breaches on all services that use this compromised identity for authentication [13].

Phishing attacks during single sign-on processes is another security concern. OpenID and other HTTP redirection-based protocols (e.g., Microsoft Live ID [56], Google AuthSub [24], AOL OpenAuth [4], Yahoo BBAuth [76]) may habituate users to being redirected to identity provider websites for authentication. If users do not verify the authenticity of these websites before entering their credentials (and they usually do not [60, 15]), phishing attacks are possible. One recent large scale empirical study of password habits found that 1.5% of users enter passwords to phishing websites annually [19].

To prevent phishing attacks, users must verify the authenticity of an identity provider before entering their credentials. Existing research on authenticating websites to users include security indicators [12, 30], secure bookmarks for known websites [14, 75, 78], and automated detection and blacklisting of known phishing sites [16]. However, studies suggest that security indicators are ineffective at preventing phishing attacks [15, 60], and blacklisting known phishing sites still suffers from a high rate of false-positives and false-negatives [79]. Even with improved security indicators, users may ignore them [74, 15, 79, 60, 64].

Sharing personally identifiable information imposes great privacy concerns. OpenID and InfoCard allow users to consent to the release of their attributes to RPs. However, studies show that users become habituated to consent forms after seeing them multiple times [23, 72, 25]. If users are asked to consent through easily dismissed mechanisms, they might simply agree if it is the most convenient way to complete their primary task [13].

Another privacy concern for Web SSOs is the issue of IdP tracking. Redirection-based Web SSOs make the tracking of browsing habits possible. For instance, in the OpenID specification, authentication related operations are indirect communications between an RP and an IdP. These operations rely on HTTP redirection (302 HTTP response) and the `return_url` field in the request header for conveying identity validation results from an IdP back to the corresponding RP. An IdP could track all the websites a user has logged into by recording data from the `return_url` field.

3.4 Lack of driving forces from IdPs

Another driving force could come from new credential validation services provided by IdPs. RPs could utilize those services to reduce their operational cost, comply with regulations, or create new business models. However, as there is no proven business model for third-party identity provision [8],

and users' privacy concerns may restrict governments and organizations from releasing their private information [34, 9], the driving forces from IdPs are too weak to drive RPs into the Web SSO ecosystem.

3.4.1 Lack of proven business model for IdPs

Developing a sustainable IdP business model is difficult. In essence, IdPs are responsible for the inaccuracy of information they provided to RPs, and they are liable for the misuse of information provided by users. IdPs are unwilling to assume the liabilities of asserting identity attributes unless they can make money doing so. To be profitable, an IdP has to demonstrate that the information it provides is valuable enough for RPs to justify the cost of collecting user-identity information themselves. Nevertheless, as RPs could simply ask their users to provide proof of their identity attributes through off-line approaches, the development of a sustainable IdP business model is limited [8]. Based on the experiences of public certification authorities, asserting identity has not yet been a profitable business [8].

The issue of trust is hindering the development of IdP business models as well. As the Web is a virtual world that is beyond the boundaries of physical governments, even a reputable organization (e.g., hospital, motor vehicle authority, police department) could and would like to provide user-attribute certification services (e.g., date of birth), RPs might not trust such an organization. Those solutions are most likely restricted to specific domains and could not scale to the Web.

Providing value-added identity services to users is another possible IdP revenue model. For instance, FreeYourID.com is an OpenID IdP that provides services for users to register a personal .name domain name (e.g. first.last.name) as an OpenID identifier. However, it is difficult to get users to use a stand-alone identity provider and pay for it. As a result, FreeYourID.com discontinued its service after two and a half years (02/2007 - 08/2009) of operation.

3.4.2 People's privacy concerns

Governments and certain organizations (e.g., banks, hospitals) already hold users' identity information. Those entities could join the Web SSO ecosystem as IdPs. However, surveys suggest that most people do not believe that their personal data is adequately protected, or that the existing laws and organizational practices provide a reasonable level of privacy protection [34, 9]. People's privacy concerns could become a negative force that prevents governments and organizations from disclosing their identity information to RPs.

4. RECOMMENDATIONS

The underlying causes of the RP adoption problem involve business incentives and concerns, usability, security, privacy, trust, and legal issues. To consolidate the broken Web SSO triangle, we suggest that future Web SSO development should (1) provide RPs with concrete business benefits, (2) address RPs' business concerns, (3) investigate IdP business models and build trust frameworks for IdPs to provide trustworthy identity-attribute assertion services, (4) balance the competition tension between IdPs and RPs for user data, and (5) build identity support into browser to facilitate RPs adoption. In the following subsections, we will discuss each recommendation in turn.

4.1 Provide RPs with concrete business gains

Web SSO solutions must provide concrete business gains on the RP side to encourage CSPs to migrate their current login procedures to Web SSO. For instance, Plaxo.com, an online address book provider, conducted a "Two-Click Sign up" experiment with Google [39] to enable Google's users (1,000 participants) to sign up and import their Google contact list into Plaxo. The result was encouraging; 92% participants completed the import task. As adopting SSO could facilitate user data import from IdPs, Plaxo was motivated to support Web SSO as an RP.

Another example that provide RPs with concrete value propositions is Facebook Connect [17]. It is a set of APIs from Facebook that enable Facebook users to log onto third-party websites, applications, mobile devices and gaming systems with their Facebook identity. In addition, Facebook Connect enables the integration of Facebook Platform directly into RPs. According to the latest Facebook statistics [18] (October 2010), more than one million RPs have integrated with Facebook Platform through Facebook Connect, and more than 150 million people engage with Facebook on different RPs every month. Compared to OpenID, Facebook Connect provides RPs with much higher business incentives. With Facebook Connect, RPs can (1) get access to users' profiles and their social graphs, (2) utilize platform-specific services such as messaging, and (3) provide a richer user experience through social plug-ins such as recommendations and activity feeds.

Unlike OpenID, which functions solely as an authentication mechanism, Facebook Connect extends Web SSO to content sharing. While being logged in, users can connect with friends via RPs, and post information and updates to their Facebook profile. This encourages more than one third of 500 million active Facebook users [18]) to use Facebook Connect as their Web SSO. As a result, the contents on the RPs' websites are exposed and propagated through the Facebook's social network, and in turn the members of Facebook may be attracted to the RP websites. And that circle of virtue is what RPs want to have.

4.2 Understand and address RPs' business concerns

Identity technology grew from within the corporate enterprise. The advantage of adopting an intra-organization SSO solution is obvious. SSO reduces operational cost and streamlines users' login experiences [49, 1]. All a SSO project needs is a cost justification for the identity management project; there are no other business concerns. Federated SSO provides mutual benefits for all organizations in the federation. Each organization can continue to manage their users while leveraging all users in the "circle of trust." As a result, all organizations in the federation are rewarded for their participation.

Unlike intra-organization and federated SSOs, Web SSOs require RPs to give up control over their customers and rely on IdPs to authenticate and assert users' attributes [13]. This change raises great business concerns. To facilitate the adoption of RPs, future Web SSO development should first investigate the concerns of CSPs about being RPs. Possible concerns are varied, and may include such categories as:

- Business needs: How can Web SSO help RPs increase their revenue and better serve their customers?

- Liability and laws: When IdPs fail, who is liable? Who should be called when customer support is needed?
- Terms and quality of service requirements for identity services: How should RPs define and validate the accuracy of identity information?
- Models for monetizing identity services: How and how much should RPs pay for the identity services provided by IdPs?
- Usability and user acceptance: How can users be provided with consistent and usable login experiences?
- Privacy: What are users' privacy concerns? How can RPs protect their privacy?

4.3 Identify IdP business models and build trust frameworks

Without a proven business model for IdPs, there won't be any valuable identity services that RPs can consume (other than asserted global identifiers). Blakley [7] suggests using *meta-identity* as a business model and a way to reduce privacy risk. In a meta-identity system, an IdP answers a RP's query using identity metadata (e.g., Bob is over 18) instead of identity data (e.g., Bob is 45). In this way, IdPs do not have to give out their critical identity data, and can minimize the disclosure of specific personal information. We recommend future research on Web SSO should investigate IdP business models that support the establishment of additional identity services.

In much the same way as credit cards reduce the friction of paying for goods and services, Web SSO systems reduce the friction of using the Web. However, one fundamental problem is that of trust: how does an RP know it can trust credentials issued from an IdP? This is a business, legal, and social problem that cannot be solved by technology. In March 2010, the Open Identity Exchange (OIX) [52] foundation was formed to build trust in the exchange of online identity credentials across public and private sectors. OIX follows an open market model to provide the certification services needed to deliver the levels of identity assurance and protection required by organizations. While OIX is a good starting point in the right direction, we suggest more effort should be invested on building diverse trust frameworks from legal, social, and business foundations.

4.4 Balance the tension between IdPs and RPs

User data is critical to the success of most CSPs. Competition between IdPs and RPs for user data is the natural cause of the Web SSO adoption problem. On one hand, IdPs need to protect their user data and treat them as valuable assets. On the other hand, RPs want to leverage the user data from IdPs as much as possible. The tension between IdPs and RPs could restrict the evolution of the Web SSO ecosystem.

To balance this tension, we suggest that the decision as to whether an RP can synchronize user data from an IdP should be based on the *ownership* of the data. For a user's personal registered or generated profile and content data, with the user's consent, the IdPs and Web SSO solution should allow RPs to access the user's personal data as easily as possible. However, for a user's social attributes (e.g., birth certificate, degrees, credit records, driving records) that are asserted by an IdP, the IdPs should act as "gate keepers" that protect the confidentiality and integrity of their customers' data and provide only meta-identity information to the RPs. By distinguishing user data this way, both IdPs and RPs can get what they need for business while users retain control over their own personal data.

4.5 Build identity support into browser

The adoption of current Web SSO solutions faces a classic chicken-and-egg problem: CSPs do not want to change their authentication procedures until a critical mass of users has adopted Web SSO, and users have little incentive to employ the technology unless many of their CSPs are supported as RPs [59]. To resolve this, future Web SSO development needs additional forces from other sources beyond the actors in the Web SSO triangle. As the browser is the central piece that communicates with all actors in the identity ecosystem, we conjecture that the browser can potentially provide a driving force for RPs to adopt SSO when the browser is directly augmented with identity support.

4.5.1 Consistent and intuitive user experience

An identity-enabled browser could provide users with a consistent and intuitive user experience. In our vision of a true Web SSO system, a user should log into her IdP once and gain access to all websites that she has an account with, without being prompted to login again on each website. In other words, when accessing N RPs using one IdP, the user should provide her credential exactly once to the IdP, consent at most N times (one for each RP if the consent was not recorded before), and should perform a logout process only once from the IdP.

Currently, most web users are not aware that they already own SSO "keys" hosted on major IdPs. To advocate users' awareness, an identity-enabled browser could prompt users to sign in before browsing, provide users with an intuitive way for selecting an identity to use when visiting websites, and make it clear that they can synchronize their personal data from IdPs to RPs right within the browser. By embedding the SSO experience into most web users' daily web-surfing activities, the browser could drive users to reach the necessary critical mass to overcome the resistance of CSPs becoming RPs.

4.5.2 Gradual engagement

A high conversion rate (i.e., the ratio of visitors who become registered users) is desirable for many websites. However, most websites convert only a fraction of visitors into customers (the average online conversion rate is around 3%, with the highest at approximately 9%) [62]. One key factor that affects a website's conversion rate is the *form abandonment rate*—the ratio of visitors that fail to complete a sign-up form [73]. Traditionally, websites redirect visitors to sign up for an account before granting them access to the protected resources or allowing them to create personal content. For password recovery, and to ensure future communication with users, most websites also require validation via email before activating an account. Many web services need to identify each individual user before providing the requested service (e.g., access to shared personal content that is controlled by an access-control-list). However, this requirement discourages potential customers from trying a new web service.

With an identity-enabled browser, an RP can avoid sign-up forms in favor of *gradual engagement*. When an anonymous visitor consents to use one of her authenticated identifiers for the visiting RP, the RP can grant the user the required permissions for the task at hand without any interruption. This instantly turns the visitor into a marketable lead, who is identifiable by the user's OpenID identifier and email address. Once the visitor is identifiable, the RP can gradually engage with the user to acquire additional attributes (e.g., gender, date of birth) when there is value for the user to provide them. Ultimately, the RP may be able to convert the user from performing actions, such as simple page browsing, to performing more desired transactions, such as sales of products or software downloads.

4.5.3 Challenges and possible solutions

Designing a usable identity-enabled browser is challenging. To be usable, the solution must leverage the skills and experiences that an average Web user already possess. It must not require that any special software be installed on end-user computers, or require users to manage public/secret keys or X.509 certificates for performing cryptographic operations. There are currently over one billion OpenID-enabled "keys" provided by major CSPs, and they are too valuable to be ignored. Thus, the solution should be backward-compatible with existing IdPs and RPs, and support gradual adoption. To facilitate adoption by RPs, the solution must not require RPs to modify their login UI. How to design a usable login UI and login interaction flow for web users is still an open problem that ongoing single sign-on research is attempting to address [58, 55]. Furthermore, the solution should be readily employable for emerging Web 2.0 applications that process personal data located on multiple CSPs.

One possible way to address the aforementioned challenges is by extending the OpenID protocol using its standard extension framework to enable the identity-enabled browser to perform mutual authentication directly with IdPs without HTTP redirection. Performing mutual authentication directly within a browser might be more intuitive for users and could reduce the chance of phishing attacks. Web users are not accustomed to using an OpenID URL as an identifier [15, 3]; email addresses on the other hand, serve as user identifiers for many CSPs [3]. To make login identifiers usable for average users, the browser could "hide" OpenID URL identifiers from users with existing email accounts by combining OpenID with an email-to-OpenID protocol such as EAUT [21] or WebFinger [27]. Once users have mutually authenticated with their IdPs, the browser could use a new HTTP access authentication scheme (similar to HTTP Basic) to convey the authenticated identities automatically into websites that support OpenID for authentication. By designing an identity-enabled browser this way, web users can authenticate with their existing email accounts directly within a browser. With the users' consent, their identities can transparently flow into OpenID-enabled websites without additional log-on steps.

To reach its maximum utility, the extended OpenID protocol should also be able to make the login process more usable in other emerging application domains. For instance, the protocol should support client-side mashups that aggregate personal data from multiple CSPs, mobile devices that have limited input capabilities, appliance devices (e.g., Netflix, XBox, Zune), and traditional rich-client applications.

5. SUMMARY

Web SSO systems pave a critical foundation for the user-centric Web where users own their personal content and are free to share it across and beyond CSPs. In our vision of a truly user-centric Web, users have the freedom to choose their favorite providers for their identities, content, social relationships, and access-control policies.

OpenID and InfoCard are mainstream user-centric solutions built to achieve single sign-on in the Web. These solutions are technically sound, but do not have a strong business model. Currently, CSPs do not have sufficient motivation to become RPs; and the driving forces from users and IdPs are not strong enough to motivate CSPs to adopt SSO. Fundamentally, Web SSO systems shift the function of identity management from RPs to IdPs. However, this change is misaligned with the business interest of an enterprise as identities are too valuable to be "shifted" out; most RPs depend on their user-base to survive. Without direct and significant returns for CSPs, it is inherently difficult to convince them to take on the business risk of being an RP.

In this paper, we identify the underlying causes behind the RP adoption problem. We use the metaphor of the broken Web SSO triangle to frame our discussion. To consolidate the triangle, we suggest recommendations from business, legal, social, and technology points of view. Learning from Facebook Connect's experience, we believe that for OpenID to succeed, new open content-sharing protocols have to be proposed that integrate user authentication with user-centric content sharing. This, in our opinion, is one way to overcome identity monopolies and gives the open user-centric Web a chance to compete and grow. Without clear value propositions for RPs and users, there is little chance that users will be able to use those billion keys when surfing the Web.

6. ACKNOWLEDGMENTS

We thank members of the Laboratory for Education and Research in Secure Systems Engineering (LERSSE) who supplied valuable feedback on the earlier drafts of this paper. We also thank Cormac Herley for shepherding this paper. Research on this work has been partially supported by the Canadian NSERC ISSNet Internetworked Systems Security Network Program.

7. REFERENCES

[1] ActivIdentity Corp. Actividentity securelogin. http://www.protocom.com/, 2009.

[2] A. Adams and M. A. Sasse. Users are not the enemy. *Commun. ACM*, 42(12):40–46, 1999.

[3] B. Adida. EmID: Web authentication by email address. In *Web 2.0 Security and Privacy Workshop 2008*, Oakland, California, USA, 2008.

[4] AOL LLC. AOL Open Authentication API. http://dev.aol.com/api/openauth, January 2008.

[5] P. Austel, S. Bhola, S. Chari, L. Koved, M. McIntosh, M. Steiner, and S. Weber. Secure delegation for web 2.0 and mashups. In *Workshop on Web 2.0 Security And Privacy*, 2008.

[6] P. Becker. Identity's First Big War: a history lesson. http://www.identityblog.com/?p=551, August 2006.

[7] B. Blakley. The meta-identity system. http://notabob.blogspot.com/2006/07/meta-identity-system.html, July 2006.

[8] B. Blakley. The Information Card Landscape. Technical report, Burton Group, Febrary 2009.

[9] CBS News. Poll: Privacy rights under attack. http://www.cbsnews.com/stories/2005/09/30/opinion/polls/main894733.shtml, October 2005.

[10] S. Chiasson, P. C. van Oorschot, and R. Biddle. A usability study and critique of two password managers. In *Proceedings of 15th USENIX UNIX Security Symposium*, pages 1–16, Vancouver, Canada, August 2-4 2006. USENIX.

[11] E. Cohen, R. K. Thomas, W. Winsborough, and D. Shands. Models for coalition-based access control (CBAC). In *Proceedings of the Seventh ACM Symposium on Access Control Models and Technologies*, pages 97–106, Monterey, California, USA, 2002.

[12] CoreStreet Ltd. Spoofstick. http://www.spoofstick.com/, 2005.

[13] R. Dhamija and L. Dusseault. The seven flaws of identity management: Usability and security challenges. *IEEE Security and Privacy*, 6:24–29, 2008.

[14] R. Dhamija and J. D. Tygar. The battle against phishing: Dynamic security skins. In *SOUPS '05: Proceedings of the 2005 Symposium on Usable Privacy and Security*, pages 77–88, New York, NY, USA, 2005. ACM.

[15] R. Dhamija, J. D. Tygar, and M. Hearst. Why phishing works. In *CHI '06: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 581–590, Montréal, Québec, Canada, 2006. ACM.

[16] Earthlink Inc. Earthlink toolbar: scambloker for Windows users. http://www.earthlink.net/, 2008.

[17] I. Facebook. Facebook Platform. http://www.facebook.com/platform, 2010.

[18] I. Facebook. Facebook Press Room Statistics. http://www.facebook.com/press/info.php?statistics, October 2010.

[19] D. Florencio and C. Herley. A large-scale study of web password habits. In *WWW '07: Proceedings of the 16th International Conference on World Wide Web*, pages 657–666, New York, NY, USA, 2007. ACM.

[20] B. Freeman. Yahoo! OpenID:One Key, Many Doors. http://developer.yahoo.com/openid/openid-research-jul08.pdf, July 2008.

[21] D. Fuelling and W. Norris. Email Address to URL Transformation 1.0. http://eaut.org/specs/1.0/, June 2008.

[22] S. Gaw and E. W. Felten. Password management strategies for online accounts. In *Proceedings of the Second Symposium on Usable Privacy and Security*, pages 44–55, 2006.

[23] N. Good, R. Dhamija, J. Grossklags, D. Thaw, S. Aronowitz, D. Mulligan, and J. Konstan. Stopping spyware at the gate: a user study of privacy, notice and spyware. In *SOUPS '05: Proceedings of the 2005 Symposium on Usable Privacy and Security*, pages 43–52, New York, NY, USA, 2005. ACM.

[24] Google Inc. Authsub authentication for web applications. http://code.google.com/apis/accounts/docs/AuthSub.html, December 2008.

[25] J. Grossklags and N. Good. Empirical studies on software notices to inform policy makers and usability designers. In *USEC 2007: Proceedings of 1st International Workshop on Usable Security*, pages 341–355, 2007.

[26] J. A. Halderman, B. Waters, and E. W. Felten. A convenient method for securely managing passwords. In *Proc. of WWW 2005*, pages 471–479, 2005.

[27] E. Hammer-Lahav. WebFinger. http://webfinger.net/, August 2009.

[28] X. Hao. Attacking Certificate-based Authentication System and Microsoft InfoCard. In *Power of Community Security Conference*, 2009.

[29] C. Herley. So long, and no thanks for the externalities: the rational rejection of security advice by users. In *NSPW '09: Proceedings of the 2009 Workshop on New Security Paradigms Workshop*, pages 133–144, New York, NY, USA, 2009. ACM.

[30] A. Herzberg and A. Jbara. Security and identification indicators for browsers against spoofing and phishing attacks. *ACM Trans. Internet Technology*, 8(4):1–36, 2008.

[31] J. Inc. RPX: User Engagement Make Easy. http://www.janrain.com/products/rpx, March 2010.

[32] Internet2. Shibboleth System. http://shibboleth.internet2.edu/, 2008.

[33] JanRain Inc. Relying Party Stats. http://www.janrain.com/blogs/relying-party-stats-april-1st-2009, 2009.

[34] K. O. Jerry DeVault, Brian Tretick. Privacy and independent verification: What consumers want. http://consumerprivacyguide.com/privacy/ccp/verification1.pdf, 2002.

[35] A. Jøsang, M. A. Zomai, and S. Suriadi. Usability and privacy in identity management architectures. In *ACSW '07: Proceedings of the Fifth Australasian Symposium on ACSW Frontiers*, pages 143–152, Darlinghurst, Australia, Australia, 2007. Australian Computer Society, Inc.

[36] B. Laurie. OpenID: Phishing Heaven. http://www.links.org/?p=187, January 2007.

[37] Liberty Alliance. Liberty Alliance Project. http://www.projectliberty.org/, 2002.

[38] E. Maler and D. Reed. The venn of identity: Options and issues in federated identity management. *IEEE Security and Privacy*, 6:16–23, 2008.

[39] J. McCrea. Introducing two-click signup. http://blog.plaxo.com/archives/2009/01/introducing_two_1.html, January 2009.

[40] C. Messina. OpenID Phishing Brainstorm. http://wiki.openid.net/OpenID_Phishing_Brainstorm, 2009.

[41] Microsoft Corp. Windows CardSpace. http://www.microsoft.com/windows/products

/winfamily/cardspace/default.mspx, 2009.

[42] D. Mills. Identity in the Browser (Mozila Labs). https://mozillalabs.com/blog/2009/05/identity-in-the-browser/, 2010.

[43] Mozila Labs. Weave Identity Account Manager. https://wiki.mozilla.org/Labs/Weave/Identity/Account_Manager, 2009.

[44] J. Mulligan and A. Elbirt. Desktop security and usability trade-offs: An evaluation of password management systems. *Information Systems Security*, 14(2):10–19, 2005.

[45] S. J. Murdoch and R. Anderson. Verified by visa and mastercard securecode: or, how not to design authentication. In *In the proceedings of Financial Cryptography and Data Security 2010*, January 2010.

[46] MyOpenID. OpenID Site Directory. http://openiddirectory.com/, 2010.

[47] A. Nanda and M. B. Jones. Identity Selector Interoperability Profile V1.5. http://informationcard.net/specifications, July 2008.

[48] Netcraft. August 2010 Web Server Survey. http://news.netcraft.com/archives/category/web-server-survey/, 2010.

[49] Novell Inc. Novell securelogin. http://www.novell.com/products/securelogin/, 2009.

[50] OASIS. Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML). http://saml.xml.org/, April 2009.

[51] OAuth Core Workgroup. Oauth core 1.0 specification. http://oauth.net/core/1.0/, December 2007.

[52] Open Identity Exchange. Building trust online identity. http://openidentityexchange.org/, March 2010.

[53] OpenID Foundation. Promotes, protects and nurtures the OpenID community and technologies. http://openid.net/foundation/, 2009.

[54] OpenID Foundation. OpenID Directory. http://openiddirectory.com/, 2010.

[55] OpenID Wiki. Openid user experience. http://wiki.openid.net/browse/view=ViewFolder&param=user-experience, April 2010.

[56] R. Oppliger. Microsoft .NET Passport and identity management. *Information Security Technical Report*, 9(1):26–34, 2004.

[57] D. Recordon and B. Fitzpatrick. OpenID authentication 2.0. http://openid.net/specs/openid-authentication-2_0.html, December 2007.

[58] E. Sachs. Usability Research on Federated Login. http://sites.google.com/site/oauthgoog/UXFedLogin, October 2008.

[59] R. Sausner. Authentication Software: Widespread Adoption Seen As Unlikely Before 2011. http://www.americanbanker.com/usb_issues/116_4/-274048-1.html, 2006.

[60] S. E. Schechter, R. Dhamija, A. Ozment, and I. Fischer. The emperor's new security indicators. In *Proceedings of the 2007 IEEE Symposium on Security and Privacy*, pages 51–65, Washington, DC, USA, 2007. IEEE Computer Society.

[61] P. Steiner. On the Internet, nobody knows you're a dog. http://en.wikipedia.org/wiki/On_the_Internet,_nobody_knows_youre_a_dog, 2010.

[62] I. Strouchliak. Conversion rate optimization. http://www.seochat.com/c/a/Website-Marketing-Help/Conversion-Rate-Optimization/, 2009.

[63] S.-T. Sun, K. Hawkey, and K. Beznosov. Secure Web 2.0 content sharing beyond walled gardens. In *Proceedings of the 25th Annual Computer Security Applications Conference (ACSAC)*, pages 409–418. ACSA, IEEE Press, December 7-11 2009.

[64] J. Sunshine, S. Egelman, H. Almuhimedi, N. Atri, and L. F. Cranor. Crying Wolf: An empirical study of SSL warning effectiveness. In *Proceedings of 18th USENIX Security Symposium*, pages 399–432, 2009.

[65] Sxipper Inc. Sxipper form manager Firefox extension. http://www.sxipper.com/, 2009.

[66] The Eclipse Foundation. Higgins Card Selectors. http://www.eclipse.org/higgins/, 2009.

[67] The Information Card Foundation. Advance the use of Information Card. http://informationcard.net/foundation, 2009.

[68] University of Michigan. Cosign: Secure, intra-institutional web authentication. http://weblogin.org/, 2009.

[69] University of Washington. Pubcookie: open-source software for intra-institutional web authentication. http://www.pubcookie.org/, 2008.

[70] VeriSign Inc. VeriSign OpenID SeatBelt Plugin. https://pip.verisignlabs.com/seatbelt.do, 2009.

[71] Wikipedia. Password fatigue. http://en.wikipedia.org/wiki/Password_fatigue, 2009.

[72] M. Wogalter. Purpose and scope of warnings. In *Handbook of Warnings*, pages 3–9. Lawrence Erlbaum Associates, 2006.

[73] L. Wroblewski. *Web Form Design: Fill in the blanks*, chapter Gradual Engagement. Rosenfeld media, 2008.

[74] M. Wu, R. C. Miller, and S. L. Garfinkel. Do security toolbars actually prevent phishing attacks? In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems(CHI '06)*, pages 601–610, New York, NY, USA, 2006. ACM.

[75] M. Wu, R. C. Miller, and G. Little. Web wallet: preventing phishing attacks by revealing user intentions. In *SOUPS '06: Proceedings of the Second Symposium on Usable Privacy and Security*, pages 102–113, New York, NY, USA, 2006. ACM.

[76] Yahoo Inc. Browser-Based Authentication (BBAuth). http://developer.yahoo.com/auth/, December 2008.

[77] Yale University. CAS: Central authentication service. http://www.jasig.org/cas, 2009.

[78] K.-P. Yee and K. Sitaker. Passpet: convenient password management and phishing protection. In *SOUPS '06: Proceedings of the Second Symposium on Usable Privacy and Security*, pages 32–43, New York, NY, USA, 2006. ACM.

[79] Y. Zhang, S. Egelman, L. Cranor, and J. Hong. Phinding phish: Evaluating anti-phishing tools. In *Proceedings of the 14th Annual Network and Distibuted System Security Symposium (NDSS 2007)*, 2007.