The Quest to Replace Passwords

The title of this article is borrowed from a fascinating research paper presented at the *2012 IEEE Symposium on Security and Privacy*. [Bonneau et al. 2012] Therein “two decades of proposals to replace text passwords for general-purpose user authentication on the web” are compared and evaluated using an explicit framework designed by the authors. Although some of the technical material in [Bonneau et al. 2012] is now somewhat dated, the framework is still quite relevant. Here we use the framework to analyze the benefits of single sign-on (SSO) with respect to ordinary password authentication. In Part 2, we apply the framework to a number of single-factor authenticators and then show how to combine them to provide two-factor authentication.

Of course one should always be careful what one asks for. In 2012, the goal was not to replace passwords (indeed, the authors argued strongly that this would not be likely to happen any time soon), but rather the quest is for an authentication technology that retains the strengths of passwords (yes, there are some) but at the same time eliminates their weaknesses. Thus the title of the paper might more accurately be:

> The quest to find a web authentication technology that is as easy to deploy and at least as usable as passwords, but significantly more secure.

The framework described in [Bonneau et al. 2012] revolves around 25 benefits. There are three categories of benefits: usability (8 benefits), deployability (6 benefits), and security and privacy (11 benefits). See section II of [Bonneau et al. 2012] for a rough interpretation of each benefit.

Contents

Summary

It is well known that **Password via SSO** is significantly more usable than **Password** alone, which is confirmed by the framework. However, if we assume that users tend to reuse a single password across all sites, we are led to the conclusion that in terms of usability, the **Password via SSO** scheme tends to legitimize existing user behavior.

The **Password via SSO** scheme exhibits marginally improved security over **Password** alone. Unfortunately, the added security comes with an apparent loss of privacy.

The **Password via SSO** scheme is not nearly as deployable as ordinary **Password**, a fact not readily made apparent by the framework. Moreover, while the framework exposes other disadvantages of **Password via SSO**, the significance of those disadvantages are only made clear upon deeper reflection and further analysis.

Reference

[Bonneau et al. 2012] Bonneau, Joseph; Herley, Cormac; Oorschot, Paul C. van; Stajano, Frank (2012). “The Quest to Replace Passwords: A Framework for Comparative Evaluation of Web Authentication Schemes.” University of Cambridge Computer Laboratory, Technical Report Number 817. Cambridge, UK. ISSN 1476-2986.

Comparing Password and Password vis SSO

By “Password via SSO” we mean password authentication in conjunction with either OpenID Connect 1.0 or SAML V2.0 Web Browser SSO, both of which are equivalent under the current framework (which says something about the framework itself). The evaluation of the **Password via SSO** scheme follows directly from the evaluation of **OpenID** in section IV-C1 in [Bonneau et al. 2012] since the same arguments apply to OpenID Connect and SAML Web Browser SSO. For convenience, those arguments are reiterated below in the context of the **Password via SSO** scheme.

For additional background, refer to section IV-C (Federated Single Sign-On) of [Bonneau et al. 2012].

Benefits Table

In the table below, all of the benefits introduced by [Bonneau et al. 2012] are used to compare the **Password via SSO** scheme with the ordinary **Password** scheme. The **Password** column is faithfully reproduced from section III of [Bonneau et al. 2012].

		Password	Password via SSO
U1	*Memorywise-Effortless*		
U2	*Scalable-for-Users*		
U3	*Nothing-to-Carry*		
U4	*Physically-Effortless*		
U5	*Easy-to-Learn*		
U6	*Efficient-to-Use*		
U7	*Infrequent-Errors*		
U8	*Easy-Recovery-from-Loss*		
D1	*Accessible*		
D2	*Negligible-Cost-per-User*		
D3	*Server-Compatible*		
D4	*Browser-Compatible*		
D5	*Mature*		
D6	*Non-Proprietary*		
S1	*Resilient-to-Physical-Observation*		
S2	*Resilient-to-Targeted-Impersonation*		
S3	*Resilient-to-Throttled-Guessing*		
S4	*Resilient-to-Unthrottled-Guessing*		
S5	*Resilient-to-Internal-Observation*		
S6	*Resilient-to-Leaks-from-Other-Verifiers*		
S7	*Resilient-to-Phishing*		
S8	*Resilient-to-Theft*		
S9	*No-Trusted-Third-Party*		
S10	*Requiring-Explicit-Consent*		
S11	*Unlinkable*		
		Password	**Password via SSO**

Comparison of Password and Password via SSO

= offers the benefit; = almost offers the benefit; no circle = does not offer the benefit.
U = usability benefit; D = deployability benefit; S = security or privacy benefit.

Portions of the above table are analyzed in detail in the sections below.

Usability Benefits

The following table is an excerpt of the complete table shown above. This abbreviated table compares the Password and Password via SSO schemes for each of the original eight (8) usability benefits defined by [Bonneau et al. 2012]:

		Password	Password via SSO

U1	*Memorywise-Effortless*		
U2	*Scalable-for-Users*		
U3	*Nothing-to-Carry*		
U4	*Physically-Effortless*		
U5	*Easy-to-Learn*		
U6	*Efficient-to-Use*		
U7	*Infrequent-Errors*		
U8	*Easy-Recovery-from-Loss*		

Comparison of Password and Password via SSO
Usability Benefits

= offers the benefit; = almost offers the benefit; no circle = does not offer the benefit.

U1. *Memorywise-Effortless*. The **Password** scheme is deemed not *Memorywise-Effortless* by the framework [Bonneau et al. 2012] since users must remember a password for each website. The **Password via SSO** scheme is rated *Quasi-Memorywise-Effortless* since most users will still have to remember a single password.

> *Users of the scheme do not have to remember any secrets at all. We grant a* Quasi-Memorywise-Effortless *if users have to remember one secret for everything (as opposed to one per verifier). [Bonneau et al. 2012]*

U2. *Scalable-for-Users*. The **Password** scheme is deemed not *Scalable-for-Users* by the framework [Bonneau et al. 2012] for the same reason the scheme is not *Memorywise-Effortless*, that is, users must remember a password for each website. OTOH, the **Password via SSO** scheme is *Scalable-for-Users* since, as noted above, users still have to remember a single password but that password can work for multiple sites.

> *Using the scheme for hundreds of accounts does not increase the burden on the user. As the mnemonic suggests, we mean "scalable" only from the user's perspective, looking at the cognitive load, not from a system deployment perspective, looking at allocation of technical resources. [Bonneau et al. 2012]*

U3. *Nothing-to-Carry*. Like the **Password** scheme, the **Password via SSO** scheme is awarded the *Nothing-to-Carry* benefit.

> *Users do not need to carry an additional physical object (electronic device, mechanical key, piece of paper) to use the scheme.* Quasi-Nothing-to-Carry *is awarded if the object is one that they'd carry everywhere all the time anyway, such as their mobile phone, but not if it's their computer (including tablets). [Bonneau et al. 2012]*

U4. *Physically-Effortless*. The **Password** scheme is deemed not *Physically-Effortless* by the framework [Bonneau et al. 2012] since a password must be typed. The **Password via SSO** scheme is *Quasi-Physically-Effortless* because passwords are typed at the identity provider only.

> *The authentication process does not require physical (as opposed to cognitive) user effort beyond, say, pressing a button. Schemes that don't offer this benefit include those that require typing, scribbling or performing a set of motions. We grant* Quasi-Physically-Effortless *if the user's effort is limited to speaking, on the basis that even illiterate people find that natural to do. [Bonneau et al. 2012]*

U5. *Easy-to-Learn*. The **Password** scheme is deemed *Easy-to-Learn* by the framework [Bonneau et al. 2012] due to years of user experience. The **Password via SSO** scheme requires one or more web user interfaces, which makes the scheme only *Quasi-Easy-to-Learn*.

> *Users who don't know the scheme can figure it out and learn it without too much trouble, and then easily recall how to use it. [Bonneau et al. 2012]*

U6. *Efficient-to-Use*. The **Password** scheme is deemed *Efficient-to-Use* by the framework [Bonneau et al. 2012] since most users type only a few characters. The **Password via SSO** scheme is *Efficient-to-Use* since the **Password** scheme has the benefit. By virtue of SSO, the **Password via SSO** scheme is probably more efficient to use than the **Password** scheme.

> *The time the user must spend for each authentication is acceptably short. The time required for setting up a new association with a verifier, although possibly longer than that for authentication, is also reasonable. [Bonneau et al. 2012]*

U7. *Infrequent-Errors*. The **Password** scheme is deemed *Quasi-Infrequent-Errors* by the framework [Bonneau et al. 2012] since typos are probable. The **Password via SSO** scheme is fully *Infrequent-Errors* by virtue of single sign-on. Once the **Password via SSO** scheme is learned, it exhibits altogether fewer errors than the **Password** scheme.

> *The task that users must perform to log in usually succeeds when performed by a legitimate and honest user. In other words, the scheme isn't so hard to use or unreliable that genuine users are routinely rejected. [Bonneau et al. 2012]*

U8. *Easy-Recovery-from-Loss*. The **Password** scheme is deemed *Easy-Recovery-from-Loss* by the framework [Bonneau et al. 2012] since passwords are easily reset. The **Password via SSO** scheme is *Easy-Recovery-from-Loss* since it is more-or-less equivalent to a password reset.

> *A user can conveniently regain the ability to authenticate if the token is lost or the credentials forgotten. This combines usability aspects such as: low latency before restored ability; low user inconvenience in recovery (e.g., no requirement for physically standing in line); and assurance that recovery will be possible, for example via built-in backups or secondary recovery schemes. If recovery requires some form of re-enrollment, this benefit rates its convenience. [Bonneau et al. 2012]*

Discussion

The **Password via SSO** scheme is *Scalable-for-Users*, which is a significant advantage. It is also *Quasi-Memorywise-Effortless* and *Quasi-Physically-Effortless*, both of which are improvements over **Password** alone. However, **Password via SSO** is only *Quasi-Easy-to-Learn*, which represents a significant new cost.

The Role of User Interfaces

As noted above, **Password via SSO** is only *Quasi-Easy-to-Learn*. This is because **Password via SSO**, by definition, implies cross-domain single sign-on, which requires user interaction at the relying party to initiate a transaction, at least in the general case involving multiple identity providers. After years of research and experimentation, the optimal identity provider discovery interface is still not in widespread use (and indeed, may have yet to be found), but it is doubtful that discovery can be eliminated altogether since such a scheme may then lack the *Requiring-Explicit-Consent* benefit (ignoring other user interfaces that may be in play).

Speaking of which, the *Requiring-Explicit-Consent* benefit needs some clarification since it's not clear what the user is consenting to according to this framework. There's a wide range of technical behaviors that might require user consent, starting with cookie-based authentication at the one end, all the way to the explicit release of Personally Identifiable Information at the other end of the spectrum. One might argue that the user should control all of this, which seems to imply multiple user interfaces.

It doesn't seem likely we will be able to avoid at least one user interface (in addition to the login interface) in a federated single sign-on scenario. Today most identity providers assert transient identifiers by default, which don't require user interaction at the identity provider, but frankly, if that's all we can expect from federation, it's probably not worth the effort. Without persistent identifiers and/or other user attributes, there's little if any opportunity for personalization or access control at the relying party. However, attribute release implies user consent of one form or the other, and in turn user consent implies a user interface.

> **User Interfaces**
>
> Federation requires one or more user interfaces but good user interface design is hard. There is still work to be done in terms of optimizing the UIs used in conjunction with single sign-on workflows.

Deployability Benefits

The following table is an excerpt of the complete table shown above. This abbreviated table compares the **Password** and **Password via SSO** schemes for each of the original six (6) deployability benefits defined by [Bonneau et al. 2012]:

		Password	Password via SSO
D1	*Accessible*		
D2	*Negligible-Cost-per-User*		
D3	*Server-Compatible*		
D4	*Browser-Compatible*		
D5	*Mature*		
D6	*Non-Proprietary*		

Comparison of Password and Password via SSO Deployability Benefits

= offers the benefit; = almost offers the benefit; no circle = does not offer the benefit.

Note that the **Password** scheme is awarded all deployability benefits by the framework. [Bonneau et al. 2012] This serves as a convenient baseline against which all other schemes are measured. See section III of [Bonneau et al. 2012] for further discussion.

D1. *Accessible*. The **Password via SSO** scheme is *Accessible* since the **Password** scheme is deemed *Accessible* by the framework.

> *Users who can use passwords are not prevented from using the scheme by disabilities or other physical (not cognitive) conditions. [Bonneau et al. 2012]*

D2. *Negligible-Cost-per-User*. All things considered, the **Password via SSO** scheme does not have *Negligible-Cost-per-User*. Although the **Password** scheme is deemed fully *Negligible-Cost-per-User* by the framework, the **Password via SSO** scheme is laden with hidden costs not exposed by the framework.

> ⚠ **Extended Deployability Benefits**
>
> For a breakdown of costs associated with the **Password via SSO** scheme, see the section on Extended Deployability Benefits below.

> *The total cost per user of the scheme, adding up the costs at both the prover's end (any devices required) and the verifier's end (any share of the equipment and software required), is negligible. The scheme is plausible for startups with no per-user revenue. [Bonneau et al. 2012]*

D3. *Server-Compatible*. The **Password via SSO** scheme is not *Server-Compatible* since a relying party deployment must conform to the SSO protocol (either OpenID Connect or SAML Web Browser SSO).

> *At the verifier's end, the scheme is compatible with text-based passwords. Providers don't have to change their existing authentication setup to support the scheme. [Bonneau et al. 2012]*

D4. *Browser-Compatible*. The **Password via SSO** scheme is *Browser-Compatible* since an ordinary off-the-shelf web browser may be used.

> *Users don't have to change their client to support the scheme and can expect the scheme to work when using other machines with an up-to-date, standards-compliant web browser and no additional software. In 2012, this would mean an HTML5-compliant browser with JavaScript enabled. Schemes fail to provide this benefit if they require the installation of plugins or any kind of software whose installation requires administrative rights. Schemes offer* Quasi-Browser-Compatible *if they rely on nonstandard but very common plugins, e.g., Flash. [Bonneau et al. 2012]*

D5. *Mature*. The **Password via SSO** scheme is *Mature* since numerous open-source implementations exist for both OpenID Connect and SAML Web Browser SSO.

> *The scheme has been implemented and deployed on a large scale for actual authentication purposes beyond research. Indicators to consider for granting the full benefit may also include whether the scheme has undergone user testing, whether the standards community has published related documents, whether open-source projects implementing the scheme exist, whether anyone other than the implementers has adopted the scheme, the amount of literature on the scheme and so forth. [Bonneau et al. 2012]*

D6. *Non-Proprietary*. The **Password via SSO** scheme is *Non-Proprietary* since both OpenID Connect and SAML Web Browser SSO are based on open published standards.

> *Anyone can implement or use the scheme for any purpose without having to pay royalties to anyone else. The relevant techniques are generally known, published openly and not protected by patents or trade secrets. [Bonneau et al. 2012]*

Discussion

There is no question that **Password via SSO** lacks the *Server-Compatible* benefit but a deeper analysis will show that the choice to federate at the relying party is not a simple one. This is especially true in the case of an existing web application that already incorporates a well-entrenched login interface, since in that case the entire authentication mechanism needs to be excised from the application. For some applications, this can be a daunting task.

Security and Privacy Benefits

The following table is an excerpt of the complete table shown above. This abbreviated table compares the **Password** and **Password via SSO** schemes for each of the original eleven (11) security and privacy benefits defined by [Bonneau et al. 2012]:

		Password	Password via SSO
S1	*Resilient-to-Physical-Observation*		
S2	*Resilient-to-Targeted-Impersonation*		
S3	*Resilient-to-Throttled-Guessing*		
S4	*Resilient-to-Unthrottled-Guessing*		

S5	*Resilient-to-Internal-Observation*		
S6	*Resilient-to-Leaks-from-Other-Verifiers*		
S7	*Resilient-to-Phishing*		
S8	*Resilient-to-Theft*		
S9	*No-Trusted-Third-Party*		
S10	*Requiring-Explicit-Consent*		
S11	*Unlinkable*		

Comparison of Password and Password via SSO
Security and Privacy Benefits

= offers the benefit; = almost offers the benefit; no circle = does not offer the benefit.

S1. *Resilient-to-Physical-Observation*. The **Password** scheme is deemed not *Resilient-to-Physical-Observation* by the framework [Bonneau et al. 2012] because even if a password is typed quickly, it can be automatically recovered from high-quality video of the keyboard. The **Password via SSO** scheme is *Quasi-Resilient-to-Physical-Observation* since this attack is only possible against a single identity provider and not for each access to a relying party.

> *An attacker cannot impersonate a user after observing them authenticate one or more times. We grant* Quasi-Resilient-to-Physical-Observation *if the scheme could be broken only by repeating the observation more than, say, 10–20 times. Attacks include shoulder surfing, filming the keyboard, recording keystroke sounds, or thermal imaging of keypad. [Bonneau et al. 2012]*

S2. *Resilient-to-Targeted-Impersonation*. The **Password** scheme is (generously) awarded the *Quasi-Resilient-to-Targeted-Impersonation* by the framework [Bonneau et al. 2012] in “the absence of user studies establishing acquaintances’ ability to guess passwords, though many users undermine this by keeping passwords written down in plain sight.” The **Password via SSO** scheme is *Quasi-Resilient-to-Targeted-Impersonation* for the same reason it is *Quasi-Resilient-to-Physical-Observation* (**S1**).

> *It is not possible for an acquaintance (or skilled investigator) to impersonate a specific user by exploiting knowledge of personal details (birth date, names of relatives etc.). Personal knowledge questions are the canonical scheme that fails on this point. [Bonneau et al. 2012]*

S3. *Resilient-to-Throttled-Guessing*. The **Password** scheme is deemed not *Resilient-to-Throttled-Guessing* by the framework [Bonneau et al. 2012] since users have a well-established poor track record with respect to password selection. The **Password via SSO** scheme is *Quasi-Resilient-to-Throttled-Guessing* for the same reason it is *Quasi-Resilient-to-Physical-Observation* (**S1**).

> *An attacker whose rate of guessing is constrained by the verifier cannot successfully guess the secrets of a significant fraction of users. The verifier-imposed constraint might be enforced by an online server, a tamper-resistant chip or any other mechanism capable of throttling repeated requests. To give a quantitative example, we might grant this benefit if an attacker constrained to, say, 10 guesses per account per day, could compromise at most 1% of accounts in a year. Lack of this benefit is meant to penalize schemes in which it is frequent for user-chosen secrets to be selected from a small and well-known subset (low min-entropy). [Bonneau et al. 2012]*

S4. *Resilient-to-Unthrottled-Guessing*. The **Password** scheme is deemed not *Resilient-to-Unthrottled-Guessing* by the framework [Bonneau et al. 2012] for the same reason the scheme is not *Resilient-to-Throttled-Guessing*. The **Password via SSO** scheme is *Quasi-Resilient-to-Unthrottled-Guessing* for the same reason it is *Quasi-Resilient-to-Physical-Observation* (**S1**).

> *An attacker whose rate of guessing is constrained only by available computing resources cannot successfully guess the secrets of a significant fraction of users. We might for example grant this benefit if an attacker capable of attempting up to 2 40 or even 2 64 guesses per account could still only reach fewer than 1% of accounts. Lack of this benefit is meant to penalize schemes where the space of credentials is not large enough to withstand brute force search (including dictionary attacks, rainbow tables and related brute force methods smarter than raw exhaustive search, if credentials are user-chosen secrets). [Bonneau et al. 2012]*

S5. *Resilient-to-Internal-Observation*. The **Password** scheme is deemed not *Resilient-to-Internal-Observation* by the framework [Bonneau et al. 2012] since a password may be intercepted and replayed. The **Password via SSO** scheme is not *Resilient-to-Internal-Observation* as malware can either steal a persistent login cookie or record the master password.

> *An attacker cannot impersonate a user by intercepting the user's input from inside the user's device (e.g., by key-logging malware) or eavesdropping on the cleartext communication between prover and verifier (we assume that the attacker can also defeat TLS if it is used, perhaps through the CA). As with* Resilient-to-Physical-Observation *above, we grant* Quasi-Resilient-to-Internal-Observation *if the scheme could be broken only by intercepting input or eavesdropping cleartext more than, say, 10–20 times. This penalizes schemes that are not replay-resistant, whether because they send a static response or because their dynamic response countermeasure can be cracked with a few observations. This benefit assumes that general-purpose devices like software-updatable personal computers and mobile phones may contain malware, but that hardware devices dedicated exclusively to the scheme can be made malware-free. We grant* Quasi-Resilient-to-Internal-Observation *to two-factor schemes where both factors must be malware-infected for the attack to work. If infecting only one factor breaks the scheme, we don't grant the benefit. [Bonneau et al. 2012]*

S6. *Resilient-to-Leaks-from-Other-Verifiers*. The **Password** scheme is deemed not *Resilient-to-Leaks-from-Other-Verifiers* by the framework [Bonneau et al. 2012] since users tend to reuse passwords across websites. The **Password via SSO** scheme is *Resilient-to-Leaks-from-Other-Verifiers* since relying parties don't store passwords.

> *Nothing that a verifier could possibly leak can help an attacker impersonate the user to another verifier. This penalizes schemes where insider fraud at one provider, or a successful attack on one backend, endangers the user's accounts at other sites. [Bonneau et al. 2012]*

S7. *Resilient-to-Phishing*. The **Password** scheme is deemed not *Resilient-to-Phishing* by the framework [Bonneau et al. 2012] since phishing is a well-known problem with passwords. The **Password via SSO** scheme is not *Resilient-to-Phishing* since a stolen password may be replayed to the identity provider at a later time. Moreover, both OpenID Connect and SAML Web Browser SSO are thought to be badly not *Resilient-to-Phishing* since each protocol involves redirection to an identity provider from a relying party.

> *An attacker who simulates a valid verifier (including by DNS manipulation) cannot collect credentials that can later be used to impersonate the user to the actual verifier. This penalizes schemes allowing phishers to get victims to authenticate to lookalike sites and later use the harvested credentials against the genuine sites. It is not meant to penalize schemes vulnerable to more sophisticated real-time man-in-the-middle or relay attacks, in which the attackers have one connection to the victim prover (pretending to be the verifier) and simultaneously another connection to the victim verifier (pretending to be the prover). [Bonneau et al. 2012]*

S8. *Resilient-to-Theft*. The **Password** scheme is deemed *Resilient-to-Theft* by the framework [Bonneau et al. 2012] since there is no physical device that can be stolen. The **Password via SSO** scheme is *Resilient-to-Theft* since the **Password** scheme has the benefit.

> *If the scheme uses a physical object for authentication, the object cannot be used for authentication by another person who gains possession of it. We still grant* Quasi-Resilient-to-Theft *if the protection is achieved with the modest strength of a PIN, even if attempts are not rate-controlled, because the attack doesn't easily scale to many victims. [Bonneau et al. 2012]*

S9. *No-Trusted-Third-Party*. The **Password** scheme has *No-Trusted-Third-Party* since no third party is involved in the transaction. [Bonneau et al. 2012] The **Password via SSO** scheme does not have the *No-Trusted-Third-Party* benefit since the identity provider is privy to each and every login.

> *The scheme does not rely on a trusted third party (other than the prover and the verifier) who could, upon being attacked or otherwise becoming untrustworthy, compromise the prover's security or privacy. [Bonneau et al. 2012]*

S10. *Requiring-Explicit-Consent*. The **Password** scheme is deemed *Requiring-Explicit-Consent* by the framework [Bonneau et al. 2012] since passwords must be typed. The **Password via SSO** scheme is deemed *Requiring-Explicit-Consent* since the protocol flow of either OpenID Connect or SAML Web Browser SSO may be augmented with an appropriate web user interface. (Google's implementation of the OpenID Connect flow is a good example of how to implement such a flow.)

Requiring-Explicit-Consent

The authors of [Bonneau et al. 2012] do not award the *Requiring-Explicit-Consent* benefit to the **OpenID** scheme. Its successors (OpenID Connect and SAML Web Browser SSO) deserve the benefit, however.

> *The authentication process cannot be started without the explicit consent of the user. This is both a security and a privacy feature (a rogue wireless RFID-based credit card reader embedded in a sofa might charge a card without user knowledge or consent). [Bonneau et al. 2012]*

S11. *Unlinkable*. The **Password** scheme is deemed *Unlinkable* by the framework [Bonneau et al. 2012] assuming each website uses an independent salt value. The **Password via SSO** scheme is deemed *Unlinkable* since both OpenID Connect and SAML Web Browser SSO specify an option for the identity provider to assert a unique user identifier per relying party.

Unlinkable

Similar to **S10**, the authors of [Bonneau et al. 2012] do not award the *Unlinkable* benefit to the **OpenID** scheme. Its successors (OpenID Connect and SAML Web Browser SSO) deserve the benefit, however.

> *Colluding verifiers cannot determine, from the authenticator alone, whether the same user is authenticating to both. This is a privacy feature. To rate this benefit we disregard linkability introduced by other mechanisms (same user ID, same IP address, etc). [Bonneau et al. 2012]*

Discussion

The **Password via SSO** scheme is *Quasi-Resilient-to-Physical-Observation*, *Quasi-Resilient-to-Throttled-Guessing*, *Quasi-Resilient-to-Unthrottled-Guessing*, and fully *Resilient-to-Leaks-from-Other-Verifiers*, all of which are improvements over **Password** alone. However, the **Password via SSO** scheme completely lacks the *No-Trusted-Third-Party* benefit, which is a significant impairment. As noted by the authors of [Bonneau et al. 2012], federated schemes have been criticized on privacy grounds.

Consolidating the Authentication Secret

As noted above, **Password via SSO** is both *Quasi-Resilient-to-Unthrottled-Guessing* and *Resilient-to-Leaks-from-Other-Verifiers*, a pair of related benefits that represent a significant improvement over **Password** alone. However, the consolidation of passwords at the identity provider has a corresponding downside (not captured by the framework), and that is, federated password stores become extremely attractive targets requiring significantly more care and protection than any simple password file at a non-federated relying party (unless you assume that most users use the same password across services, in which case a simple password may be just as valuable as a federated password).

In any case, effective federation requires a quantifiable level of assurance (LoA) at the identity provider. Without some form of LoA, a non-federated service provider is hesitant to federate in the first place, and rightly so. Every successful federation is based on trust that assures service providers that identity provider(s) can be trusted to do the Right Thing. This notion is reinforced by the *No-Trusted-Third-Party* benefit, which of course is lacking in the case of **Password via SSO**.

> **Level of Assurance**
>
> To effectively replace **Password** with **Password via SSO**, a corresponding level of assurance at the identity provider is needed. Indeed, there can be no effective federation without LoA.

Making Room for Improvement

A number of security and privacy benefits stay the same as a result of single sign-on. In particular, both **Password** and **Password vis SSO** lack the *Resilient-to-Phishing* benefit, an important security benefit. Indeed phishing remains an extremely important open problem in practice. We consider this issue in more detail below and in the sequel.

Extending the Framework

Extended Password

In section III of [Bonneau et al. 2012], the authors note that “our ratings of passwords and other schemes do assume that poor user behavior is an inherent aspect of fielded systems.” The authors go on to declare that the **Password** authentication scheme is not *Resilient-to-Leaks-from-Other-Verifiers* since the framework assumes users reuse passwords across sites. Okay, that makes sense, but shouldn’t the same assumption apply to usability as well?

That rhetorical question is not considered by [Bonneau et al. 2012] and so we hazard a guess why poor user behavior with respect to usability is ignored. It seems that user behavior that increases usability at the expense of security is treated conservatively by the framework, that is, the former is ignored while the latter is incorporated into the framework’s assessment. Such an interpretation would explain the benefits awarded to the **Password** scheme in section III of [Bonneau et al. 2012].

In any case, it’s instructive to re-evaluate the usability of the **Password** scheme assuming users reuse a single password across all sites. Let’s called this the **Extended Password** scheme:

		Password	Extended Password	Password via SSO
U1	*Memorywise-Effortless*			
U2	*Scalable-for-Users*			
U3	*Nothing-to-Carry*			
U4	*Physically-Effortless*			
U5	*Easy-to-Learn*			
U6	*Efficient-to-Use*			
U7	*Infrequent-Errors*			
U8	*Easy-Recovery-from-Loss*			

Comparison of Three Password Schemes
Usability Benefits

= offers the benefit; = almost offers the benefit; no circle = does not offer the benefit.

When users reuse a single password across all sites, usability is improved as shown in the above mini-table. The **Extended Password** scheme is *Quasi-Memorywise-Effortless*, *Scalable-for-Users*, and fully *Infrequent-Errors*, all of which are improvements over the **Password** scheme. The **Extended Password** scheme is not *Physically-Effortless*, however, since the password must be typed for each authentication.

Now compare these results with the **Password via SSO** scheme, which also benefits from a single password across all sites. Except for *Physically-Effortless* and *Easy-to-Learn*, the usability benefits of **Extended Password** and **Password via SSO** are identical.

Usability of Password via SSO

In terms of usability, the **Password via SSO** scheme tends to legitimize existing user behavior.

Extending the Benefits

Extended Deployability Benefits

A number of important deployability benefits are omitted from this framework, such as *Negligible-Setup-Cost* and *Negligible-Support-Costs*. The latter is extremely important when considering a new (or enhanced) authentication technology. The **Password** scheme is deemed *Negligible-Setup-Cost* by the extended framework (as a baseline cost for comparison purposes) but it does not have *Negligible-Support-Costs* as is well known. OTOH, the **Password via SSO** scheme does not have *Negligible-Setup-Cost* benefit but it is thought to have at least *Quasi-Negligible-Support-Costs*.

Related to this, the *Negligible-Cost-per-User* benefit is not granular enough and too broad for our purposes. Subtracting out the previous pair of cost-related benefits, the original *Negligible-Cost-per-User* benefit could be broken down into two separate benefits: *Negligible-Fixed-Cost-per-User* and *Negligible-Variable-Cost-per-Login*. In the sequel, some schemes have the former but not the latter; whereas other schemes have just the opposite. These are important considerations as it turns out.

		Password	Password via SSO
D7	*Negligible-Setup-Cost*		
D8	*Negligible-Support-Costs*		
D9	*Negligible-Fixed-Cost-per-User*		
D10	*Negligible-Variable-Cost-per-Login*		

Extended Deployability Benefits

= offers the benefit; = almost offers the benefit; no circle = does not offer the benefit.

Extended Security Benefits

Two important benefits not captured by the original framework (which was published in 2012) are *No-Shared-Secrets-on-the-Server* and *Resilient-to-Verifier-Impersonation*. The **Password** scheme is *Quasi-No-Shared-Secrets-on-the-Server* since passwords are vulnerable to offline attacks. Likewise, **Password via SSO** is *Quasi-No-Shared-Secrets-on-the-Server* but for a different reason: even though the number of password files is reduced to one, that one file becomes an extremely attractive target, with value equal to the sum of the individual password files required in the absence of federation.

		Password	Password via SSO
S12	*No-Shared-Secrets-on-the-Server*		
S13	*Resilient-to-Verifier-Impersonation*		

Extended Security Benefits

= offers the benefit; = almost offers the benefit; no circle = does not offer the benefit.

Recall that neither the **Password** scheme nor the **Password via SSO** scheme is *Resilient-to-Phishing*. This benefit will become extremely important in the sequel. For example, schemes that involve one-time passwords (OTPs) are inherently *Resilient-to-Phishing* according to the framework's definition of the benefit. Yet OTPs (and other schemes) are not resilient to active man-in-the-middle-attacks, which is cause for increasing concern in 2019.

Going beyond *Resilient-to-Phishing*, we introduce the *Resilient-to-Verifier-Impersonation* benefit. Since neither **Password** nor **Password via SSO** is *Resilient-to-Phishing*, certainly neither can be awarded the *Resilient-to-Verifier-Impersonation* benefit. The latter becomes important in the sequel, however.