FOM Hochschule für Oekonomie & Management

Hochschulzentrum Düsseldorf

Project Paper

part-time degree program

6. Semester

in the study course "Wirtschaftsinformatik"

as part of the course

Web & Social Media Analytics

on the subject

Sign in with Google - Opportunities and risks of using single sign-on providers for managing user accounts on websites

by

Luis Pflamminger

Advisor:	Prof. Dr. Carolin Tewes
Matriculation Number:	538276
Submission:	22. August 2022

Contents

List of Figures **IV**

List of Abbreviations **V**

1 Introduction **1**

1.1 Abstract 1
1.2 Motivation 1
1.3 Problem Definition and Goal 2
1.4 Approach 2

2 Fundamentals of Single Sign-On (SSO) **2**

2.1 Basics of Authentication 2
2.1.1 Authentication & Authorization 2
2.1.2 Authentication flows without SSO 3
2.2 Single Sign-On and Federated Identity 3
2.2.1 Definitions 3
2.2.2 Types & Use-Cases of SSO 4
2.2.3 Web SSO Protocols & Technologies 5
2.3 Web SSO Providers 9
2.3.1 Overview 9
2.3.2 Sign In With Google 9

3 Opportunities **11**

3.1 Higher Conversion Rates 11
3.2 Simpler Account Management 12
3.3 Increased Security 13
3.4 Cross-Platform User Tracking 13

4 Risks of using SSO Services **14**

4.1 Association with providers 14
4.2 Loss of Users 14
4.3 Leakage of User Data 14
4.4 Technical Complexity of Integration 14

5 Conclusion **14**

Bibliography **15**

Internet sources **16**

List of Figures

Figure 1: Flow chart of SFA authentication process 3
Figure 2: Types of Single Sign-On 5
Figure 3: Flow of SAML Protocol 6
Figure 4: Flow of OpenID Protocol 7
Figure 5: Flow of OAuth 2.0 Protocol 8
Figure 6: Sign In With Google UI 10

List of Abbreviations

SSO	Single Sign-On
SMS	Short Message Service
SFA	Single-Factor Authentication
2FA	Two-Factor Authentication
MFA	Multi-Factor Authentication
API	Application Programming Interface
IdP	Identity Provider
SP	Service Provider
RP	Relying Party
FIM	Federated Identity Management
SAML	Security Assertion Markup Language
US	United States of America

1 Introduction

1.1 Abstract

1.2 Motivation

One of the most difficult aspects of operating a website or app is dealing with user authentication and passwords.

Regularly, hackers try to break into the databases of internet services to reveal account data and passwords of their users.

The best-practices for encrypting passwords and storing them safely are changing continuously and keeping up with new security standards and requirements is costly and time consuming.

Managing passwords and usernames is not only a technical challenge, but also puts a burden on end users.[1]

A majority of users still don't manage their passwords in the recommended way, which is generating a new password for each login and storing every password in a secure password safe. A modern internet user on average uses x different services that require a login. This results in most users using the same password for many or all of these services, which makes it easy for hackers to overtake a users online presence if only one of their passwords is hacked or leaked.

Because of these problems, SSO is good fivehead.

Digital services have revolutionized the way people

Remembering passwords is hard bla bla

Reusing the same password is insecure and hackers have used this

- Account management is hard - Hackers are trying to find security flaws and publish passwords - Two factor authentication is annoying for customers etc.

[1] Cp. *Hoonakker*, *P.*, *Bornoe*, *N.*, *Carayon*, *P.*, 2009, p. 459.

1.3 Problem Definition and Goal

1.4 Approach

2 Fundamentals of SSO

2.1 Basics of Authentication

2.1.1 Authentication & Authorization

Authentication is the act of establishing a users identity. The user has to prove, that he is who he says he is[2]. There are three main ways how a user can prove his identity:

1. The user provides some secret, that only he knows, e.g. a password.
2. The user provides something, that only he has. An example for this is an Short Message Service (SMS) based authentication system, where the user receives a message with a code, which has to be entered into the login form. He thus proves, that he is in possession of the phone with the phone number he has previously provided.
3. The user proves his identity by using some unique physical characteristic, e.g. a fingerprint or eye retina scan.[3]

Using only one of these ways to authenticate a user is called Single-Factor Authentication (SFA), using two is called Two-Factor Authentication (2FA) and using multiple factors is called Multi-Factor Authentication (MFA). Using two or more factors for authentication improves security and is deemed essential for applications with sensitive data.[4]

Contrary to authentication, authorization determines, wether a user is allowed to access a specific resource. It happens once the user has been authenticated.[5] For example, user A might make a post on a social media site, which is only shared with his close friends. If another user B wants to view the post, the system first checks if user B is part of user A's close friends. If he is, he is authorized to see the resource.

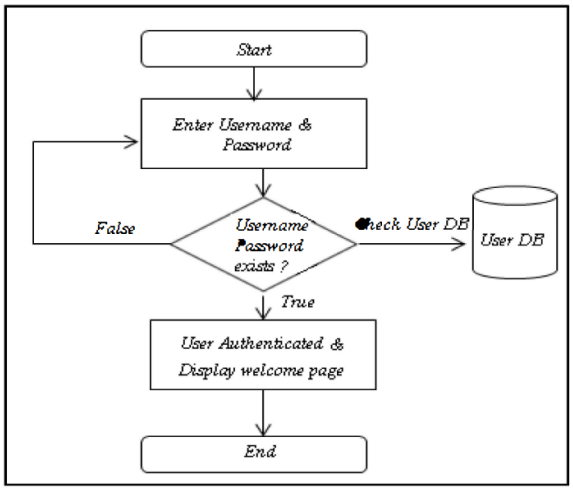
---

[2] Cp. *Basavala*, *S. R.*, *Kumar*, *N.*, *Agarrwal*, *A.*, 2012, p. 398.
[3] Cp. ibid., p. 398.
[4] Cp. *Drew*, *J.*, 2019.
[5] Cp. *Auth0 Inc.*, n. d.

2.1.2 Authentication flows without SSO

Figure 1 shows a flow chart for a traditional authentication flow using a username and password. The user enters his information and clicks the "Sign In" button. Next, the server checks in the database, wether or not the username exists and the password is correct. If this is true, the user is authenticated and he can enter the website.[6]

Figure 1: Flow chart of SFA authentication process



Source: *Basavala*, *S. R.*, *Kumar*, *N.*, *Agarrwal*, *A.*, 2012

Using 2FA introduces complexity into this process, as the user now has to additionally provide something he has. In the case of SMS 2FA there are now multiple databases needed to store passwords, phone numbers and the codes sent via SMS[7]. Additionally, software needs to be operated which generates the codes and sends them to the user. Managing this complexity and ensuring security for all components is both time- and cost-intensive.

2.2 Single Sign-On and Federated Identity

2.2.1 Definitions

SSO is a mechanism which allows users to sign into multiple independent software systems using a single set of credentials[8]. It also only requires the user to perform a single action to authenticate against multiple participating services. Such systems or services could

[6] Cp. *Basavala*, *S. R.*, *Kumar*, *N.*, *Agarrwal*, *A.*, 2012, p. 400.
[7] Cp. *Basavala*, *S. R.*, *Kumar*, *N.*, *Agarrwal*, *A.*, 2012, p. 400.
[8] Cp. *Radha*, *V.*, *Reddy*, *D. H.*, 2012, p. 134.

be apps, websites or technical interfaces like Application Programming Interfaces (APIs). After signing in, the user is not asked to enter their password again when visiting a different service[9]

SSO is made possible by an **Identity Provider (IdP)**, which provides a central server for authentication. When talking about SSO, the website operator is usually refered to as **Service Provider (SP)**. The users authenticate with the IdP and it shares the user's identity with the SP.[10] The SP does not confirm the user's identity in any way, so he has to trust the IdP to correctly deliver user identities[11]. Originally, SSO could only be used by members of a single organization to sign into different applications like HR software, payroll or communication systems, because there were no open standards, that allowed companies to share identities with other organizations.[12]

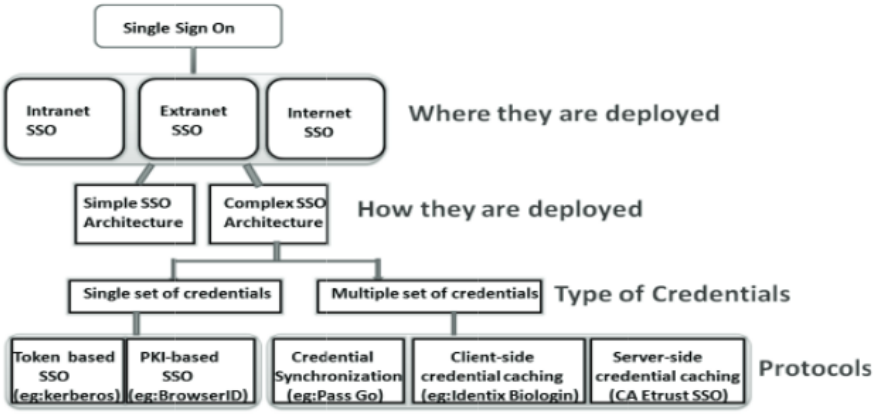
Federated Identity Management (FIM) is the broader term for managing user identities across apps and websites from different companies. It provides standards which companies can use to share user identities between trusted domains, which allows SSO to be deployed not just in organizations, but also on the open web. These standards are what allows end-users to use IdPs like Google and Facebook to sign into many different websites.[13]

2.2.2 Types & Use-Cases of SSO

Figure 2 shows the different types of SSO. Intranet SSO is used within the secure network of an organization and allows its members to access multiple applications with one set of credentials. This is relatively easy to deploy, as all components and clients are administrated by the same organization, eliminating the need for open standards and trust to third parties. Extranet SSO connects applications and SPs from different organizations. This is the basis for Web SSO, which is the type this paper is concerned with. It is based on web technologies and allows users to access different websites with a single set of credentials.[14]

[9] Cp. *Bazaz*, *T.*, *Khalique*, *A.*, 2016, p. 18.
[10] Cp. *Beltran*, *V.*, 2016, p. 24.
[11] Cp. *Nallathamby*, *J.*, 2018.
[12] Cp. *Okta*, n. d.
[13] Cp. ibid.
[14] Cp. *Radha*, *V.*, *Reddy*, *D. H.*, 2012, p. 135.

Figure 2: Types of Single Sign-On



Source: *Radha*, *V.*, *Reddy*, *D. H.*, 2012

2.2.3 Web SSO Protocols & Technologies

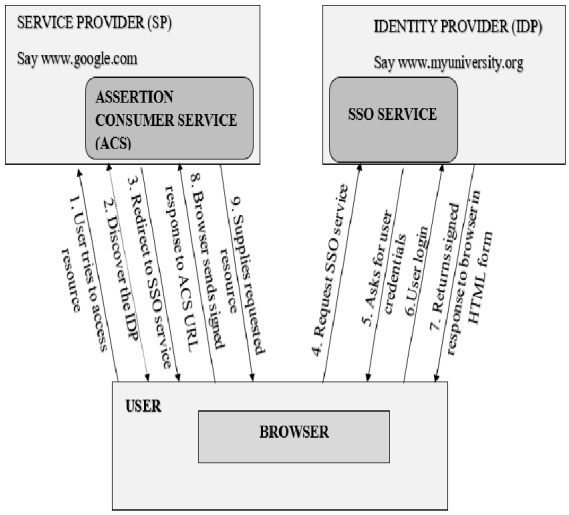
There are many different protocols defined for SSO. Some are only used for Intranet SSO, which is not covered in this paper. The next section will therefore only cover the three most used Web SSO protocols[15].

Security Assertion Markup Language

Security Assertion Markup Language (SAML) is an open standard for exchanging user identities and authorization information between an IdP and a service provider. It uses the markup language XML for communication between applications. The typical authentication flow using SAML is shown in figure 3. An SP contacts an IdP and requests a user identity. The SAML standard does not specify how the IdP has to authenticate the user, but only defines the communication between the two parties. After the user has authenticated with the IdP the identity information is sent back to the SP. It includes, whether the user is authenticated, what roles and rights they have and which data and resources they are allowed to access.[16] Note that while the SP basically communicates directly with the IdP, all communication still goes through the user. Which IdP is used is predefined by the SP, the user does not get to choose where they enter their credentials or who provides their identity. The SP has to trust the IdP completely, because it also provides authorization information.

[15] Cp. *OneLogin*, n. d.
[16] Cp. *Radha*, *V.*, *Reddy*, *D. H.*, 2012, p. 137.

Figure 3: Flow of SAML Protocol



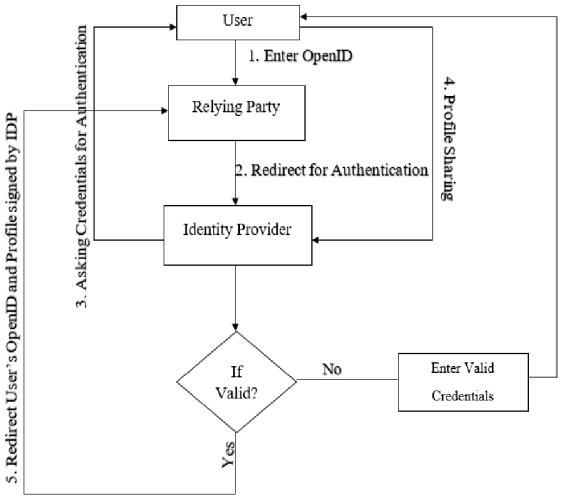
Source: *Bazaz*, *T.*, *Khalique*, *A.*, 2016, p. 21

OpenID

This protocol works differently than SAML, in that the user gets to choose who provides their identity. They might use a large IdP like Google or Facebook or even set up their own OpenID service. The SP is called the Relying Party (RP) in this model. The IdP is only responsible for authentication of users and providing identity. It does not deliver any other authorization information.[17] The flow is shown in figure 4 and works like this. The user visits the website and clicks the sign in button. They are then prompted to enter their OpenID identity provider. The RP (so the website) then redirects the user to the site of the IdP, where the user authenticates by entering their credentials. In the next step the user tells the IdP, which information they want to share with the RP. If the credentials are valid, the selected identity information is passed to the RP, otherwise the user is asked to enter valid credentials.[18] This seperation of RP and IdP and introduction of user choice is the basis for Web SSO as we know it, where websites allow users to choose their preferred identity provider. SAML does not allow for this, as the IdP is predefined by the SP.

[17] Cp. *Bazaz*, *T.*, *Khalique*, *A.*, 2016, p. 21.
[18] Cp. ibid., p. 21.

Figure 4: Flow of OpenID Protocol



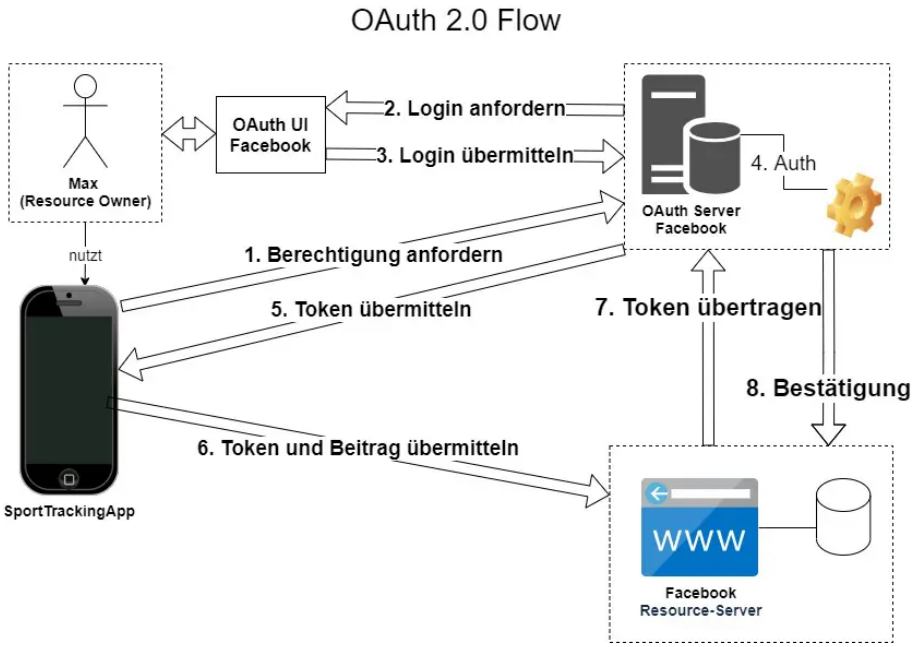
Source: *Bazaz*, *T.*, *Khalique*, *A.*, 2016, p. 22

OAuth 2.0 & OpenID Connect

This SSO protocol differentiates from the others, in that it centers around authorization rather than authentication. An example use-case for this is a third-party app wanting to create a Facebook post in the user's name on their profile. For this, the app does not need an identity, but rather the permission from Facebook to create a post. It contacts Facebook's OAuth Server and asks it for permission. Facebook then contacts the user which owns the resource (in this case the Facebook account) and asks them for permission. They accept by logging into their account and accepting access from the app. The Facebook OAuth Server sends back a token to the app which can be used to create the post. The app then contacts Facebook's Resource Server to actually create the post, providing the token to prove it is authorized.[19]

[19] Cp. *Wesener*, *M.*, 2021.

Figure 5: Flow of OAuth 2.0 Protocol



Source: *Wesener*, *M.*, 2021, p. 22

While OAuth 2.0 is a protocol specification, OpenID Connect is a concrete implementation of the protocol. The terms "OpenID" and "OpenID Connect" are different and can't be used interchangably.[20]

To further understand the difference between the OAuth 2.0 und OpenID functionalities consider an e-commerce example, where the user has to enter their payment information. They have previously entered it into their Google account and would like to automatically fill the information from Google into the shop.

Using OpenID, the shop would use Google as an IdP to get the user's identity and automatically create a corresponding user account on the website. As OpenID simply provides the identity and can't provide the payment information from Google, the user would still have to enter their payment information manually. It would then be saved in the user's account on the web shop (not Google) and when he returns to buy a different item, the information could be retrieved from the SP's server.

With OAuth, the SP would not have to create a user account using the identity from Google and would not have to store any information on its server. It would simply request access to the payment information resource from Google. The user would authenticate and allow the information transfer. Then the SP could retrieve the information from Google and use it in the checkout process.

[20] Cp. *Wesener*, *M.*, 2021.

2.3 Web SSO Providers

2.3.1 Overview

There are many SSO solutions targeted at businesses for internal use. In addition to an IdP service they often provide an identity management suite which allows businesses to easily deploy SSO within their network. The solutions targeting end users on the open web are often called "Social Login" services, as they rely on social networks that have a wide userbase like Facebook, Twitter, Google or LinkedIn.

The most popular social networks worldwide are Facebook with 2.9 billion and YouTube with 2.6 billion monthly active users[21]. As YouTube is based on user accounts from Google, the social login offerings of Google and Facebook are examined.

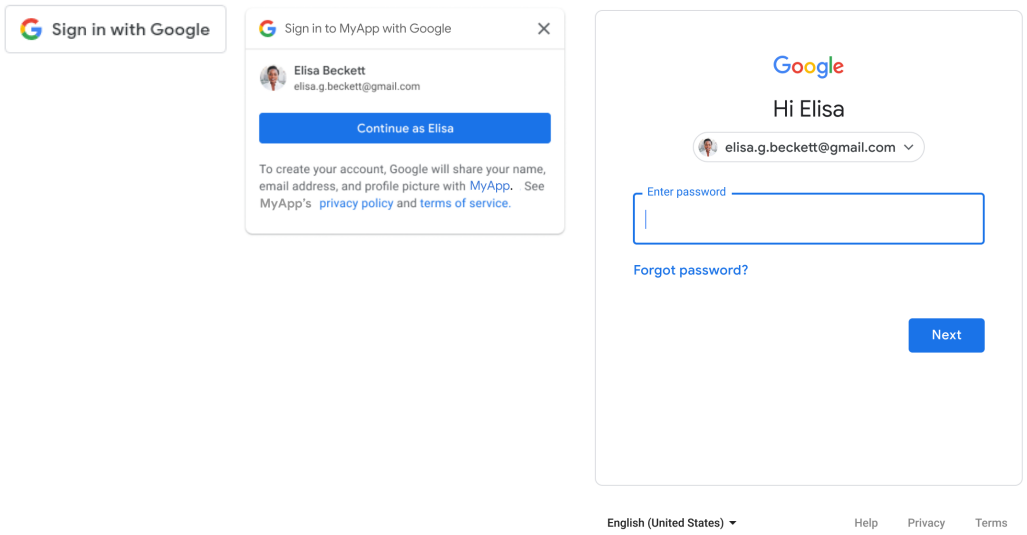
2.3.2 Sign In With Google

Google's service is based on the OAuth 2.0 protocol. This ensures a flexible service, as SPs can create a seperate account on their server but do not have to, as they can retrieve required resources using OAuth.[22] Customizable buttons are provided by Google which redirect the user to Googles sign in page (figure 6a). Google claims data from the feature is not used for advertising or other non-related purposes. If users are already signed into their Google account when visiting the website, Google offers the possibility to sign in with one click using a pop up, as shown in figure (figure 6b). Google offers an additional autorization interface to allow loading other data from a users Google account.[23]

[21] Cp. *DataReportal, We Are Social, Hootsuite*, 2022.
[22] Cp. *Google*, 2022.
[23] Cp. ibid.

Figure 6: Sign In With Google UI

(a) Sign In With Google Button **(b)** One-Tap Sign In **(c)** Google Password Prompt Source:



Google, 2022

Integrating the Sign In With Google service into an existing website is fairly easy. SPs have to create a project on Google's developer website. They then have the option to configure the consent screen shown to the user by Google and enter e.g. their website name and logo, a support e-mail adress and which user data the application wants to retrieve.[24]

Website developers can integrate Google's authentication interface into the website with just a few lines of code. After the user authenticates, the user data is delivered into the web application and can be used from there. If SPs have existing account management infrastructure, integrating this method might certainly require more time, as data might come in a different format or database structures might not line up.[25]

[24] Cp. *Google*, 2022.
[25] Cp. ibid.

3 Opportunities

3.1 Higher Conversion Rates

The conversion rate is one of the most important e-commerce metrics. It gives the percentage of users who visited the website that resulted in actual sales.[26] There are a number of factors that can influence conversion rate. On the one hand these include multiple aspects that have nothing to do with the website. The type and range of products offered on the website are important. A store selling clothing might have a much higher conversion rate than a website selling luxury cars, for example[27]. The pricing strategy also has an impact. If the website is offering attractive pricing on their products and additional bonuses like free shipping or frequent discounts, conversion rates are increased[28]. The promotion aspect is another important factor. If the brands advertising clearly an effectively promotes its products visitors are more likely to buy on the first visit[29].

The factors mentioned above are related to the web presence, in that offers like discounts and free shipping need to be effectively presented on the platform. But in order to have a decent conversion rate, technical factors on the website also have to be addressed[30].

Usability is directly related to how the website is built and which features the e-commerce system has[31]. There are three dimensions of usability on a website. Information Quality describes how accurate, relevant and timely the content on the website is. Service Quality describes how interactive and responsive the website is, how good the search function works and how well the security and privacy policies are presented. System Quality includes how easy the site is to navigate, how fast the checkout process is and aspects like accessibility, and consistency of layout.[32] Research has shown, that all three dimensions have a direct impact on conversion rates and customer retention. Users, which perceive a higher information and system quality are significantly more likely to complete a purchase[33]. Service quality might have a measurable impact, but studies diverge here[34,35]. A survey shows that the second most common reason why adults in the United States of America (US) do not complete the checkout process, is that they had to create an account.

[26] Cp. *Gabir*, *H. H.*, *Karrar*, *A. Z.*, 2018, p. 1.
[27] Cp. *Fatta*, *D. D.*, *Patton*, *D.*, *Viglia*, *G.*, 2018, p.165.
[28] Cp. ibid., p.165.
[29] Cp. ibid., p.165.
[30] Cp. *Gabir*, *H. H.*, *Karrar*, *A. Z.*, 2018, p. 5.
[31] Cp. ibid., p. 2f.
[32] Cp. *Kuan*, *H. H.*, *Bock*, *G.-W.*, *Vathanophas*, *V.*, n. d., p. 3.
[33] Cp. ibid., p. 6.
[34] Cp. ibid., p. 6.
[35] Cp. *Gabir*, *H. H.*, *Karrar*, *A. Z.*, 2018, p. 5.

17% said, that the checkout process was too long and complicated.[36] Additional research shows, that a requirement as simple as entering the e-mail address has a high negative impact on conversion rate[37]

Web SSO offers solutions to all factors which increase conversion rates mentioned above. By signing in with an IdP before or during the checkout process, users do not have to create a seperate account with the e-commerce site. Even for web shops, that don't require users to create an account, SSO offers benefits. Traditionally users have had to re-enter their personal information like name, address, phone number and birth date with each site they were shopping on[38]. With SSO users can choose to share their information that they have registered with the IdP and thus skip the process of entering their information. Thus, SSO increases the checkout speed both with required account creation and without. This also directly addresses two common reasons why customers abandon their cart during the checkout process.

3.2 Simpler Account Management

User account management, authentication and authorization are big parts of web development and require a lot of development resources. Using SSO, the burden of implementing user authentication forms, developing password criteria, storing passwords securely, designing appropriate databases and many more aspects are completely put on the IdP. Furthermore, providers like Google and the well defined SSO protocols make implementation of social login services easy for developers. This helps to reduce development resources, complexity and cost.[39]

In addition to development, the administration of a website is simplified when using SSO. User data stored with the IdP does not need to be managed by the SP, reducing storage needs operation complexity and cost. Of course, the degree of savings depends on what percentage of users is willing to adopt SSO and wether or not traditional login is still needed[40]. As shown in section 2.3.2, IdPs like Google make it possible to completely abandon account management and only use Google's identity management service. Depending on the needs of the SP, this approach might limit the functionality of the website[41].

[36] Cp. *Baymard Institute*, 2022.
[37] Cp. *McDowell*, *W. C.*, *Wilson*, *R. C.*, *Kile*, *C. O.*, 2016, p. 4.
[38] Cp. *Beltran*, *V.*, 2016, p. 24.
[39] Cp. *Bazaz*, *T.*, *Khalique*, *A.*, 2016, p. 22.
[40] Cp. ibid., p. 22.
[41] Cp. ibid., p. 22.

3.3 Increased Security

As users that use social login don't have to create passwords for each website they have an account with, security is increased. On average, users seem to have between six and seven unique passwords. Each of these passwords is reused on just under six different sites, which totals an average of about 36 unique login combinations that each user has to remember.[42] Reusing the same password on multiple sites is very insecure. If a website is compromised and passwords are leaked to the public, hackers can try the same username and password combination on other websites to gain access. Additionally, if a user is subject to a phishing attack, hackers are able to compromise multiple of the user's accounts.[43] Even if a website is not responsible for a password leak, users hacked this way will still associate a negative experience with the brand and reduce interactions in the future. SSO eliminates this issue, as users do not need to think of new passwords for every login and are therefore not able to reuse the same password twice. As IdPs are usually large corporations, their security standards regarding credentials are strict and breaches are less likely to happen. **(source)**

3.4 Cross-Platform User Tracking

[42] Cp. *Florencio*, *D.*, *Herley*, *C.*, 2006, p. 4.
[43] Cp. *McDade*, *M.*, 2022.

4 Risks of using SSO Services

4.1 Association with providers

4.2 Loss of Users

4.3 Leakage of User Data

4.4 Technical Complexity of Integration

5 Conclusion

Table summarizing results. Only looked at it from a technical perspective. Focused on scenario, that website is developed by the company itself. Most of the points covered in this paper do not applied to websites hosted using services like Shopify or Wix. Even if self developed, there might be frameworks that drastically reduce the complexity of account management. Even then opportunities like the increased conversion rate make a good argument for using Web SSO.

Bibliography

Basavala, *Sreenivasa Rao*, *Kumar*, *Narendra*, *Agarrwal*, *Alok* (2012): Authentication: An overview, its types and integration with web and mobile applications, in: 2012 2nd IEEE International Conference on Parallel, Distributed and Grid Computing, s.l.: IEEE, 2012-12

Bazaz, *Tayibia*, *Khalique*, *Aqeel* (2016): A Review on Single Sign on Enabling Technologies and Protocols, in: International Journal of Computer Applications, 151 (2016), Nr. 11, pp. 18–25

Beltran, *Victoria* (2016): Characterization of web single sign-on protocols, in: IEEE Communications Magazine, 54 (2016), Nr. 7, pp. 24–30

Fatta, *Davide Di*, *Patton*, *Dean*, *Viglia*, *Giampaolo* (2018): The determinants of conversion rates in SME e-commerce websites, in: Journal of Retailing and Consumer Services, 41 (2018), pp. 161–168

Florencio, *Dinei*, *Herley*, *Cormac* (2006): A Large Scale Study of Web Password Habits, tech. rep. MSR-TR-2006-166, s.l., 2006-11, p. 10, URL: https://www.microsoft.com/en-us/research/publication/a-large-scale-study-of-web-password-habits/

Gabir, *Hamim Hamid*, *Karrar*, *Azza Z.* (2018): The Effect of Website's Design Factors on Conversion Rate in E-commerce, in: 2018 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE), s.l.: IEEE, 2018-08

Hoonakker, *Peter*, *Bornoe*, *Nis*, *Carayon*, *Pascale* (2009): Password Authentication from a Human Factors Perspective: Results of a Survey among End-Users, in: Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 53 (2009), Nr. 6, pp. 459–463

Kuan, *Huei Huang*, *Bock*, *G.-W.*, *Vathanophas*, *V.* (n. d.): Comparing the Effects of Usability on Customer Conversion and Retention at E-Commerce Websites, in: Proceedings of the 38th Annual Hawaii International Conference on System Sciences, s.l.: IEEE

McDowell, *William C.*, *Wilson*, *Rachel C.*, *Kile*, *Charles Owen* (2016): An examination of retail website design and conversion rate, in: Journal of Business Research, 69 (2016), Nr. 11, pp. 4837–4842

Radha, *V.*, *Reddy*, *D. Hitha* (2012): A Survey on Single Sign-On Techniques, in: Procedia Technology, 4 (2012), 2nd International Conference on Computer, Communication, Control and Information Technology(C3IT-2012) on February 25 - 26, 2012, pp. 134–139

Internet sources

Auth0 Inc. (n. d.): Authentication vs. Authorization, <https://auth0.com/docs/get-started/identity-fundamentals/authentication-and-authorization> (no Date) [Access: 2022-08-20]

Baymard Institute (2022): 48 Cart Abandonment Rate Statistics 2022, <https://baymard.com/lists/cart-abandonment-rate> (2022) [Access: 2022-08-21]

DataReportal, We Are Social, Hootsuite (2022): Most popular social networks worldwide as of January 2022, ranked by number of monthly active users (in millions), <https://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/> (2022-01-06) [Access: 2022-08-21]

Drew, *Jeremy* (2019): Two-factor authentication becomes mandatory for many online transactions, <https://www.lexology.com/library/detail.aspx?g=d036cdcc-2f99-4da8-837f-9b1b5a08366f> (2019-07-04) [Access: 2022-08-20]

Google (2022): Sign In With Google, <https://developers.google.com/identity/gsi/web/guides/overview> (2022-07-28) [Access: 2022-08-21]

McDade, *Mirren* (2022): 5 Reasons To Avoid Password Reuse, <https://expertinsights.com/insights/5-reasons-you-should-never-reuse-passwords/> (2022-01-25) [Access: 2022-08-21]

Nallathamby, *Johann* (2018): What is Federated Identity Management?, <https://wso2.com/articles/2018/06/what-is-federated-identity-management/> (2018-06-18) [Access: 2022-08-21]

Okta (n. d.): Worin unterscheiden sich föderierte Identitätsverwaltung (FIM) und Single Sign-On (SSO)?, <https://www.okta.com/de/identity-101/federated-identity-vs-sso/> (no Date) [Access: 2022-08-20]

OneLogin (n. d.): Technologies Used in Federated Identity, <https://www.onelogin.com/learn/federated-identity> (no Date) [Access: 2022-08-21]

Wesener, *Maximilian* (2021): OAuth 2.0 – Ein Überblick, <https://blog.doubleslash.de/oauth-2-0-ein-ueberblick/> (2021-03-10) [Access: 2022-08-21]

Declaration in lieu of oath

I hereby declare that I produced the submitted paper with no assistance from any other party and without the use of any unauthorized aids and, in particular, that I have marked as quotations all passages which are reproduced verbatim or near-verbatim from publications. Also, I declare that the submitted print version of this thesis is identical with its digital version. Further, I declare that this thesis has never been submitted before to any examination board in either its present form or in any other similar version. I herewith agree that this thesis may be published. I herewith consent that this thesis may be uploaded to the server of external contractors for the purpose of submitting it to the contractors' plagiarism detection systems. Uploading this thesis for the purpose of submitting it to plagiarism detection systems is not a form of publication.

Düsseldorf, 21.8.2022

(Location, Date)

(handwritten signature)