LITERATURE SURVEY ON SINGLE SIGN-ON FOR AUTHENTICATION

KAI-CHIEH HUANG (112676186), NISHANT SHANKAR (112670702),
PRATEEK NARENDRA (112687058), RAVINDER SINGH (112681203)

STONY BROOK UNIVERSITY

Abstract. User authentication is vital for the secure functioning of any IT system. Though there have been multiple methods proposed for authentication, simple passwords remain the dominant method for authentication. This indicates that other methods lack certain aspects that make it less appealing than password-based authentication. But a method most desirable that has benefits over other techniques is a single sign-on mechanism. A single sign-on method allows using a single credential to gain access to multiple systems. This way a user needs to remember a single credential instead of a host of credentials. In this survey, we review the various single sign-on methods and evaluate them in terms of usability, cost and user acceptance.

1 INTRODUCTION

Authentication, proving the identity of a user, has been the Achilles's heel of security research. Despite decades of efforts on multiple methods for user authentication, password based methods are still dominant and prevalent. Password based systems are a nightmare for both security researcher and users alike. Research has proved that password based systems is teeming with problems. From the perspective of an implementer, user passwords stored on a system need to be protected from illegal access which involves sophisticated methods of protection. Any error in these methods would result in the password of users being leaked or stolen [1]. From the perspective of the user, as we have access to multiple accounts, the number of credentials to remember increases. This makes the user to reuse passwords that allows attackers to use simple dictionary attacks to compromise a system. A study has shown that 51% of the users reuse their passwords across multiple platforms and some make simple edits to their passwords, which are easy to identify[2].

Among multiple systems for authentication, Single Sign-On (SSO) systems are gaining traction. SSO systems allow a user to login to multiple independent systems using a single credential, where the single credential acts as an authentication token to multiple systems. SSO offers mobility, using a single credential a user can access multiple services on different platforms[3]. An SSO mechanism reduces the burden on the user to remember multiple credentials, which to a large extent solves the problem of password reuse. The remainder of the article discusses the various protocols used that are part of the SSO system.

2 LITERATURE SURVEY

2.1 SAML

Security Assertion Markup Language (SAML) is an XML based standard used for authentication across multiple web domains and is used as a way to implement the single-sign-on model for identity management. The SAML suite[4] provides a variety of profiles to achieve SSO, but all of them rely on XML and implicitly assume the use of web domains. The architecture follows a request/response flow wherein the Service Provider (SP) requests for identity information about a user from an Identity Provider (IP). Trust between domains is established using certificates/keys and provider-specific metadata. The flow of SAML is ideally what we want in authentication systems, but in practice has poor interoperability with RESTful APIs and mobile applications. Due to its mature ecosystem and web support, it is mainly implemented in enterprise level software to allow employee access to different web domains through an SSO. Although there exist some solutions that extend SAML to work on native apps, it is relatively cumbersome to deploy, and therefore consumer directed SSO options tend to avoid SAML.

2.2 OAuth/OpenID

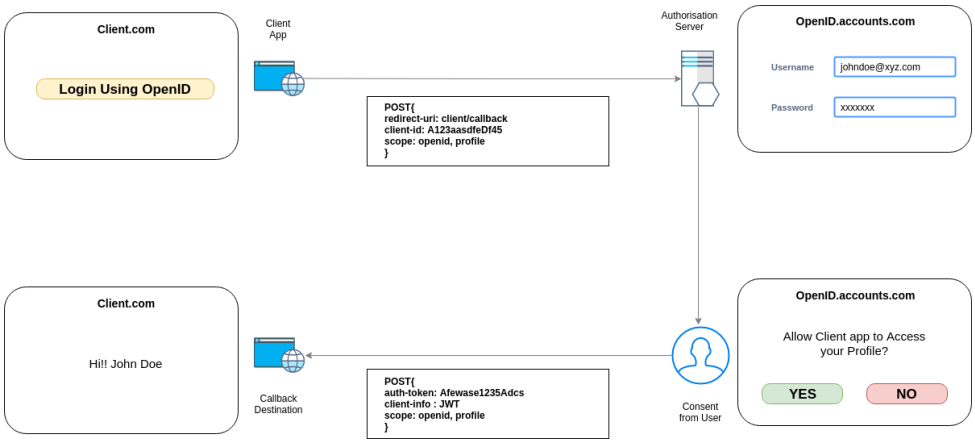

Fig. 1. Implicit flow of OpenID

While OAuth was designed for authorization to data access, it was not designed for authentication purpose. However, many applications did use OAuth for authentication by writing some wrappers over it. The reason that OAuth shouldn't be used for authentication is because there is no standard way for getting user information and it doesn't have a common set of scopes for access. To

resolve these issues with minimal overhead, researchers came up with OpenID Connect. It is not a separate standard or a protocol. It is a small overhead over OAuth 2.0 for authentication. The additions for OpenID Connect are

1. Basic information of the user (ID Token)
2. 'userInfo' endpoint for getting more user information
3. Standard set of scopes
4. Standardized implementation

The flow of OpenID Connect is exactly the same as OAuth. Except that in the initial call to the Authorization server, it adds 'openid' to the scope. And ID token is returned by the authorization server which the application can use to get basic user details[5].

With an OpenID system the user only needs to remember only a single master credential, hence limiting the number of credentials required. It is also easy for users to adapt, as the interface resembles the simple login/password systems. OpenID systems are mature and have been adopted by multiple platforms, with a strong developer community. Though OpenID seems like a holy grail solution, it is susceptible to information leak from within i.e. the owners of the credentials have complete access to other systems the user has granted access.

2.3 Self-Sovereign Identity

Self-Sovereign Identities(SSI) is a system that allows users to create and own their identities. This system enables the user to be the sole owner of his/her identity and can share his personal information at his/her discretion. These systems are being portrayed as alternatives to Federated SSO systems as these systems remove the need for a centralized authority[6]. The proponents of SSI cite problems like data breaches [7], non-ownership of the users identity and the mass analysis of the user's personal information for marketing[8] as the main evils of federated systems. To overcome the drawbacks, SSI employs decentralization using a blockchain system. The user's personal information is disseminated over a set of peers in an encrypted format. On request by the user, the information is retrieved from the system and presented to any willing entity that needs authentication[9].

Though a blockchain-based SSI system can be implemented for authentication (uport, ShoCard and BlockAuth)[9], their intent is mostly for authorization of resources and applications involving Know Your Customer (KYC). These systems involve extensive use of cryptography keys that requires special storage mechanism, that may involve a hardware token or special software. These systems are are also infrastructure-heavy they require a substantial cost to set up and operate. A SSI based system has a steep learning curve and cannot be easily taught to the an average user. The current implementation of these systems cater to specific use cases of authentication and will need to develop further to be made mainstream[10].

References

1. Robert Morris and Ken Thompson. Password security: A case history. *COMMUNICATIONS OF THE ACM*, 22:594–597, 1979.
2. Anupam Das, Joseph Bonneau, Matthew Caesar, Nikita Borisov, and Xiaofeng Wang. The tangled web of password reuse. 01 2014.
3. Vedala Radha and D. Reddy. A survey on single sign-on techniques. *Procedia Technology*, 4:134–139, 12 2012.
4. Conor Cahill, John Aol, Atos Hughes, Hal Origin, Bea Lockhart, Michael Systems, Beach, Rebekah Boeing, Booz Metz, Rick Hamilton, Booz Randall, Hamilton Allen, Irving Wisniewski, Hewlett-Packard Reid, Paula Austel, Maryann Ibm, Hondo, Michael Ibm, Mcintosh, and Trustgenix. Profiles for the oasis security assertion markup language (saml) v2. 0. 01 2004.
5. Openid implicit flow. https://openid.net/specs/openid-connect-implicit-1_0.*html*.
6. Quinten Stokkink and J.A. Pouwelse. Deployment of a blockchain-based self-sovereign identity, 06 2018.
7. Jim Isaak and Mina Hanna. User data privacy: Facebook, cambridge analytica, and privacy protection. *Computer*, 51:56–59, 08 2018.
8. Julia Woolley, Anthony Limperos, and Mary Beth Oliver. The 2008 presidential election, 2.0: A content analysis of user-generated political facebook groups. *Mass Communication and Society*, 13:631–652, 11 2010.
9. Shu Yun Lim, Pascal Tankam Fotsing, Abdullah Almasri, Omar Musa, Miss Laiha Mat Kiah, Tan Fong Ang, and Reza Ismail. Blockchain technology the identity management and authentication service disruptor: A survey. *International Journal on Advanced Science, Engineering and Information Technology*, 8(4-2):1735–1745, 2018.
10. Uwe Der, Stefan Jähnichen, and Jan Sürmeli. Self-sovereign identity − opportunities and challenges for the digital revolution, 2017.