Rethinking Single Sign-On: A Reliable and Privacy-Preserving Alternative with Verifiable Credentials

Athan D. Johnson
Department of Computer Science
University of Kentucky
Lexington, KY, USA
athan.johnson@uky.edu

Ifteher Alom
Department of Computer Science
University of Kentucky
Lexington, KY, USA
ifteheralom@uky.edu

Yang Xiao
Department of Computer Science
University of Kentucky
Lexington, KY, USA
xiaoy@uky.edu

ABSTRACT

Single sign-on (SSO) has provided convenience to users in the web domain as it can authorize a user to access various resource providers (RPs) using the identity provider (IdP)'s unified authentication portal. However, SSO also faces security and privacy challenges including single-point failure of IdP and identity linkage and profiling of users. In this paper, we present the initial design of an alternative SSO solution called VC-SSO that elevates SSO's security and privacy while preserving usability. VC-SSO leverages the recently emerged decentralized identifier (DID) and verifiable credential (VC) framework in that a user only needs to authenticate with the IdP once to obtain a VC and then may generate multiple verifiable presentations (VPs) from the VC to access different RPs. This is based on the design that each RP has established a smart contract with the IdP encoding their service agreement which includes a credential schema specifying the minimal subset of VC attributes to include in the VP. We hope the proposed VC-SSO design marks the first step toward a future SSO system that provides strong reliability and privacy to users under decentralized, adversarial settings.

CCS CONCEPTS

• **Security and privacy → Authentication**; **Privacy-preserving protocols**; **Security protocols**.

KEYWORDS

Single sign-on, privacy, authentication, verifiable credential

ACM Reference Format:
Athan D. Johnson, Ifteher Alom, and Yang Xiao. 2023. Rethinking Single Sign-On: A Reliable and Privacy-Preserving Alternative with Verifiable Credentials. In *Proceedings of the 10th ACM Workshop on Moving Target Defense (MTD '23), November 26, 2023, Copenhagen, Denmark*. ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3605760.3623767

1 INTRODUCTION

The single sign-on (SSO) authentication scheme has provided users with great convenience in accessing various resources on the Web. It allows a user to authenticate with one identity provider (IdP) in order to get authorized to use the service or resource from multiple resource providers (RPs). SSO relieves users from the hassle of curating one set of login credentials for each RP, representing an efficient and centralized paradigm of access management. For example, one can sign up for different travel apps with their Google account; a university student can log into Zoom Meetings or Microsoft Office apps by authenticating at the university portal. The classical SSO protocols date back to the late 90s. Since 2014, most popular SSO services have gradually transitioned to the OpenID Connect (OIDC) protocol [14] which relies on OAuth 2.0 [7], an authorization framework allowing a web application to access resources hosted by other web applications on behalf of a user.

Despite the above benefits and wide adoption, recent studies have shown that SSO is not immune to security, reliability, and privacy risks. First of all, the fact that SSO relies on a central entity, i.e., the IdP, for managing user credentials and providing authentication service in one place risks a single-point failure [11, 16]. The IdP has to be online to authenticate users and generate authentication tokens. Loss of IdP availability can result in failed access to the RPs under the SSO service. Second, SSO allows a curious IdP to track a user's access across different RPs without the user's explicit consent. The access data acquired by IdP from RPs along with the initial registration data provided by the user may help the IdP profile the user, posing a privacy threat [9]. In a high-risk scenario, a compromised IdP may leak a user's credentials and authentication tokens, potentially causing identity theft and unauthorized access to RPs. From an attacker's perspective, the reliance on a central IdP for authentication signifies the attack value. Third, SSO is also a target for a malicious outsider or a group of colluding RPs. The attacker may monitor the user's traffic to the IdP and RP websites with consequences in privacy violation. For instance, after successfully authenticating a user, the IdP sends an authentication token to the corresponding RP which contains statements such as user identity, timestamp, expiry, and recipient RP [14]. The colluding RPs may combine different tokens to generate a user profile, creating heightened risks of linkage attack [6, 11, 17].

Our Contribution. In this work, we present our vision and initial design of an alternative SSO mechanism that is robust against unreliable IdPs and protects user privacy. The scheme, dubbed VC-SSO, takes advantage of Verifiable Credentials (VCs) and Decentralized Identifiers (DIDs) which are being standardized by the W3C [18, 19]. VC, in a nutshell, is a digital information piece containing claims about (or attributes of) a user subject and cryptographically signed by the credential issuer. DID is the critical identity management framework that allows users to curate their own identities and store credential schemas for VCs. It relies on a Verifiable Data Registry, such as a blockchain. As for our scheme, instead of relying on the IdP for authenticating each sign-on, VC-SSO requires

MTD '23, November 26, 2023, Copenhagen, Denmark

ACM ISBN 979-8-4007-0256-3/23/11.
https://doi.org/10.1145/3605760.3623767

users to authenticate with the IdP (now the VC Issuer) just once and obtain a VC specifying the user's DID and a list of attributes that qualify the user for a range of RP services. When requesting access to a target RP, the user generates a Verifiable Presentation (VP) containing a derived VC and shows it to the RP. To limit the information leakage to RPs, the derived VC discloses the minimally required set of attributes of the original VC. The derived VC critically leverages the Zero-Knowledge Proof (ZKP) technique to show the validity of the hidden attributes and the user's knowledge of the original VC's signature. The ZKPs are indistinguishable from a random number and thus make different VPs unlinkable to each other. Once a VP is verified, a secure channel between the user and the RP can be established for service provisioning.

To be compatible with the incumbent SSO service model, VC-SSO should allow users to access different RPs, each may ask for a different set of attributes to authorize the user. We accommodate this leveraging the smart contract capability of the underlying DID-supported blockchain. The IdP establishes a smart contract with each RP which encodes the service agreement and a credential schema specifying which attributes are needed to pass RP authorization. The user may observe the credential schema and generate the VP accordingly. As a side benefit, the public schema also enables individual users to control privacy loss in an informed manner.

In summary, we make the following contributions in this paper:

- We identify the security and privacy challenges facing the existing SSO model, particularly the IdP single point of failure and privacy risk.
- We introduce VC-SSO, an alternative SSO scheme based on DID and VC, providing strong guarantees of SSO service reliability (tolerating IdP downtime) and user privacy.
- We discuss the outstanding issues toward making VC-SSO as usable as existing SSO schemes.

2 BACKGROUND

2.1 SSO Basics

While there have been various SSO schemes, modern-day SSO schemes commonly adopt the OpenID Connect protocol (OIDC) [14] which is built on top of OAuth 2.0 [7]. OAuth (Open Authorization) 2.0 is an authorization protocol standardized by IETF and designed to allow web application users to access resources from different RPs using a single set of credentials. It uses access tokens from an IdP as a means to authorize users to different RPs. OIDC extends OAuth 2.0 to an authentication protocol. It uses JSON Web Tokens (JWT) format to supply user data requested by RPs [14], allowing the latter to verify the identity of the user and to obtain user profile information. A high-level SSO scheme between a user, the IdP, and an RP works as follows. The IdP and RP are assumed to have a pre-established service agreement so that certain users under IdP are eligible for access to the RP. First, the user starts a login session on RP and provides his identifier or username that specifies the domain of IdP. Second, RP redirects the user to IdP. The user passes authentication at IdP which sends back an authentication token that includes information like user identity, intended RP, token expiration, and IdP signature [10]. Third, the user provides the authentication token to RP who then verifies it before approving the user for service.

2.2 SSO Security and Privacy Issues

In the SSO paradigm, the IdP is solely responsible for the storage, maintenance, and verification of user credentials, making IdP a high-value target and a significant single point of failure [11, 16]. Given that IdP itself operates honestly, it may still suffer from loss of availability which is nonetheless mandatory for a successful SSO process. For the SSO protocol, recent studies also show that the OIDC protocol faces implementation-level flaws including IdP confusion and malicious endpoints [10].

Compared to the risk of single-point failure, the privacy characteristic of SSO is a relatively newer area of research. Uruena et al. [17] point out how sensitive user information could be compromised from the URL parameters and HTTP Referrer header and SSO providers like Facebook Connect could easily track user activity. Morkonda et al. [12] present an empirical analysis of the privacy issues in OAuth-based SSO services of four major IdP, Google, Facebook, Apple, and LinkedIn. Their results highlight that at least one of the categories of personal data that services require is undeniably privacy-intrusive. Also, privacy-friendly login options tend to be listed towards the end. Wang et al. [20] also analyzed the major SSO and identity providers and reported several vulnerabilities including modification of identity and unauthorized access. Meanwhile, RPs are also potential privacy violators. The colluding RPs may combine different authentication tokens to generate a user profile, creating heightened risks of linkage attack [6, 11, 17]. Flaws in the RP architecture concerning storage, relaying, and validation of authentication tokens can also expose user credentials and enable unauthorized access [12, 15]. In general, there is limited scope for users to control the disclosure policy of their personally identifiable information (PII) to RPs.

2.3 Existing Solutions

Eyeing the potential exposure of SSO authentication tokens, Zhou et al. [23] propose a framework, SSOScan, for vulnerability testing by simulating attacks and monitoring traffic. SPRESSO [4] defines a protocol that aims to improve user privacy by making sign-in sessions on one SP indistinguishable from another. Lin et al. [8] propose a user-controlled SSO mechanism where users can generate a session key to establish a secure communication window for accessing different telemedicine providers. UPPRESSO [6] uses temporary pseudo-identity to access RPs in order to facilitate a privacy-preserving SSO experience, reducing inter-RP linkability.

Relevant to our work, [9, 22] explored the possibilities of incorporating self-sovereign identity (SSI) into SSO and federated identity management. [9] envisions using DID and VC to give more user control over personal data in the existing OIDC-based SSO where the IdP is still needed for VC verification in an online fashion. In comparison, we seek to minimize IdP's involvement in the authentication (i.e., not required to be always online) in order to address the availability issue.

3 SYSTEM DESIGN

3.1 Preliminaries: DID, VC, VP

Decentralized Identifiers (DID) are standardized by the W3C as a means to enable Self Sovereign Identity (SSI) management and to

establish user control and autonomy over their digital identity [3]. The DID, an identifier string, is user-controlled and curated along with other verification information including the user's public key in a DID document. The DID document is stored on a Verifiable Data Registry (VDR) which is often fulfilled by a blockchain system. The blockchain has been widely recognized for its distributed database and computing capabilities where all transactions occurring in the network are kept in a verifiable and irreversible manner [21]. Users may curate their DID documents by interacting with the VDR through blockchain transactions.

A Verifiable Credential (VC) is a set of claims about a subject, cryptographically signed by an *issuer* certifying the integrity of the claims. VCs are reminiscent of the credentials constantly used in daily lives, such as driver's licenses, medical insurance cards, etc. The VCs are stored in digital wallets (also called holders) and can be presented by the user in full or partial as a Verifiable Presentation (VP), and can be verified by a verifier by resolving the DIDs and signatures. The notion of SSI thus decouples digital identity schemes from passwords, centralized registries, identity providers, etc., making users the sole owner and controller of his/her identification and personal attributes in the web [18].

Verifiable Presentation (VP) is a subset of claims from one or more VC issued by one or more issuers and provides the user with the freedom of *selective disclosure*, that is publishing parts of the claims to verifiers instead of the whole set of claims. A VP is a tamper-evident presentation encoded in such a way that authorship of the data can be trusted after a process of cryptographic verification. VP may also contain certain data that is synthesized from the original verifiable credential(s). To release the above properties, VP incorporates the concept of Zero Knowledge Proof (ZKP) that realizes the selective disclosure property [19]. We are specifically interested in the commit-and-prove ZKP system (CP-ZPK) [2, 5, 13] where a cryptographic commitment of a secret is proved without having the prover opening the secret later.

3.2 System Model

The VC-SSO system comprises five types of entities: issuer, resource provider (RP), user, digital wallet, and blockchain.

Issuer provides initial authentication and issues VCs to users. It can be repurposed by an IdP of the SSO model, such as Google, Facebook, or a university's IT department. It has corporate agreements with different RPs. For example, a university may purchase a group subscription to an RP's web service, specifying that any university or student may access the service with no extra charge.

RP is an entity that provides web services and resources to users according to its corporate service agreement with the issuer. It may ask for a credential schema that specifies the required attributes for a user to access its resource. It also provides authenticated users with session tokens to improve the SSO user experience.

User obtains a VC from an issuer and aims to access web resources from different RPs. The VC makes the user the sole controller of his credentials and personal data and at the same time provides cryptographic proof of the authenticity and integrity of the credential.

Wallet is a user-trusted application hosted in a user client (e.g., web browser) that can store, verify, and generate VPs from VCs.

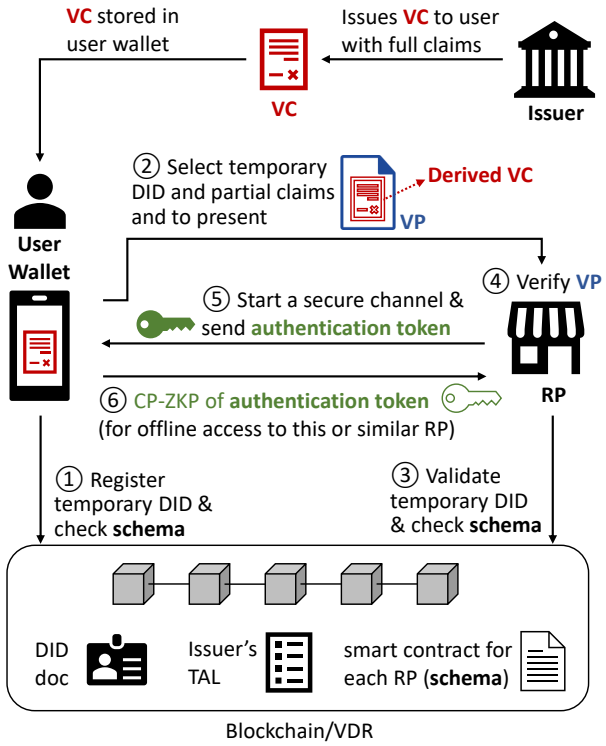

Figure 1: VC-SSO High-level Architecture

The wallet also generates a DID that links user identity in the VC to the blockchain. In this way, the user credentials, personal data, and access control are managed by the user rather than a third-party organization. In addition, the wallet also provides a mechanism to store authentication tokens and generate cryptographic commitments of tokens for quick access in subsequent sessions to RP.

Blockchain serves as the **VDR** that stores the DID documents and issuer-RP service agreements as an immutable, persistent ledger. It provides a platform for VC issuers to manage a list of their trusted RPs (called Trusted Anchor List (TAL) in SSO terminology). Each issuer-RP pair co-manages a smart contract to encode their service agreement that encompasses the credential schema (i.e., a list of attributes for accessing RP's services) required by the RP as well as executing issuer-RP business logic to support SSO.

3.3 Workflow

In our proposed VC-SSO, a typical user access request is processed in several interactions between the system components. Figure 1 shows the system workflow between Issuer, User/Wallet, RP, and the blockchain. We assume User has obtained a VC from Issuer who has an agreed schema requirement with RP which is encoded in a smart contract. The following procedures are involved to enable RP access to User:

- User opens the login page of the RP and opts for the VC-SSO option. RP then initiates a new login request and prompts the digital wallet to provide identity proof.
- User wallet generates a temporary random DID, submits it to the blockchain, and checks the required scheme (step ①).

- User wallet generates a VP and submits it to the RP (step ②). It encapsulates a derived VC containing the temporary DID, required partial claims per the schema, and a ZKP to prove the knowledge of the original VC.
- After receiving a VP, RP checks the displayed DID and retrieves the DID's public key from the corresponding DID document and VP schema from the blockchain (step ③).
- RP verifies the VP and grants User proper access rights (step ④). The success also helps generate an authentication token and establish a secure channel that is used for delivering the token and provisioning RP service to the User (step ⑤).
- For the next time the User wants to access RP (the previous secure channel is closed), User presents a commit-and-proof of the token with ZKP (CP-ZKP) to RP, which will establish a new secure channel for service (step ⑥).

3.4 Security Analysis

Threat Model. We assume users are trustworthy. The issuer follows the determined protocol but may suffer from downtimes—it may not always be available for contact by a user or RP. The issuer is also curious about which RPs a user has accessed. An RP is motivated to provide the correct resource to a user as long as the user has required verifiable attributes. It however is curious about user information beyond what is required.

First of all, since the VP presentation and verification process does not involve the issuer (IdP), the issuer's downtime does not affect RP-side user authorization. This addresses the single-point failure problem from the availability perspective and also the curious IdP problem. Second, the selective disclosure property of VPs enabled by the ZPK mechanism minimized the user information leaked to the RP. User's knowledge of the hidden attributes is validated with the ZKP in VP. The ZKPs are also indistinguishable from a random string and do not provide useful information for linkage attacks across access sessions to different RPs. A curious outsider will not know if a specific user has access to certain RPs. Lastly, the CP-ZKPs of the authentication token are also indistinguishable from a random string which prevents an outsider from tracking a user throughout different access sessions to the same RP.

4 CONCLUSION

We proposed a vision and initial design for a Verifiable Credential (VC)-based single sign-on (SSO) mechanism called VC-SSO. VC-SSO addresses some of the most critical security and privacy challenges of existing SSO services, namely the single-point failure of the identity provider and the privacy risk of identity linkage. VC-SSO takes advantage of a blockchain-enabled Decentralized Identifier (DID) platform to store an updatable credential schema and the Zero-Knowledge Proof (ZPK) technique to allow a user to generate a Verifiable Presentation (VP) from their original credential exposing minimum requirement information. In the future, we plan to implement a fully functioning VC-SSO system in the web setting. We will leverage Hyperleger Indy as the blockchain platform and BBS+ signatures [1] to efficiently construct the VCs and ZKPs. We will also explore incorporating a revocation mechanism to allow the issuer to reign in the SSO process if the user's access right should be terminated before the VC's expiry.

ACKNOWLEDGMENT

This work was supported in part by the US National Science Foundation under grant number 2247561.

REFERENCES

[1] Man Ho Au, Willy Susilo, and Yi Mu. 2006. Constant-size dynamic k-TAA. In *Security and Cryptography for Networks: 5th International Conference, SCN 2006, Maiori, Italy, September 6-8, 2006. Proceedings 5*. Springer, 111–125.
[2] Ran Canetti, Yehuda Lindell, Rafail Ostrovsky, and Amit Sahai. 2002. Universally composable two-party and multi-party secure computation. In *Proceedings of the thiry-fourth annual ACM symposium on Theory of computing*. 494–503.
[3] Md Sadek Ferdous, Andrei Ionita, and Wolfgang Prinz. 2022. SSI4Web: A Self-sovereign Identity (SSI) Framework for the Web. In *International Congress on Blockchain and Applications*. Springer, 366–379.
[4] Daniel Fett, Ralf Küsters, and Guido Schmitz. 2015. Spresso: A secure, privacy-respecting single sign-on system for the web. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. 1358–1369.
[5] Jens Groth, Rafail Ostrovsky, and Amit Sahai. 2006. Perfect non-interactive zero knowledge for NP. In *Advances in Cryptology-EUROCRYPT 2006: 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, St. Petersburg, Russia, May 28-June 1, 2006. Proceedings 25*. Springer, 339–358.
[6] Chengqian Guo, Jingqiang Lin, Quanwei Cai, Wei Wang, Fengjun Li, Qiongxiao Wang, Jiwu Jing, and Bin Zhao. 2021. UPPRESSO: Untraceable and Unlinkable Privacy-PREserving Single Sign-On Services. *arXiv preprint arXiv:2110.10396* (2021).
[7] Dick Hardt. 2012. *The OAuth 2.0 authorization framework*. Technical Report.
[8] Tzu-Wei Lin, Chien-Lung Hsu, Tuan-Vinh Le, Chung-Fu Lu, and Bo-Yu Huang. 2021. A smartcard-based user-controlled single sign-on for privacy preservation in 5G-IoT telemedicine systems. *Sensors* 21, 8 (2021), 2880.
[9] Zoltán András Lux, Dirk Thatmann, Sebastian Zickau, and Felix Beierle. 2020. Distributed-ledger-based authentication with decentralized identifiers and verifiable credentials. In *2020 2nd Conference on Blockchain Research & Applications for Innovative Networks and Services (BRAINS)*. IEEE, 71–78.
[10] Christian Mainka, Vladislav Mladenov, Jörg Schwenk, and Tobias Wich. 2017. SoK: single sign-on security—an evaluation of openID connect. In *2017 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE, 251–266.
[11] Eve Maler and Drummond Reed. 2008. The venn of identity: Options and issues in federated identity management. *IEEE security & privacy* 6, 2 (2008), 16–23.
[12] Srivathsan G Morkonda, Sonia Chiasson, and Paul C van Oorschot. 2021. Empirical analysis and privacy implications in OAuth-based single sign-on systems. In *Proceedings of the 20th Workshop on Workshop on Privacy in the Electronic Society*. 195–208.
[13] Torben Pryds Pedersen. 1991. Non-interactive and information-theoretic secure verifiable secret sharing. In *Annual international cryptology conference*. Springer, 129–140.
[14] Natsuhiko Sakimura, John Bradley, Mike Jones, Breno De Medeiros, and Chuck Mortimore. 2014. Openid connect core 1.0. *The OpenID Foundation* (2014), S3.
[15] San-Tsai Sun and Konstantin Beznosov. 2012. The devil is in the (implementation) details: an empirical analysis of OAuth SSO systems. In *Proceedings of the 2012 ACM conference on Computer and communications security*. 378–390.
[16] San-Tsai Sun, Eric Pospisil, Ildar Muslukhov, Nuray Dindar, Kirstie Hawkey, and Konstantin Beznosov. 2011. What makes users refuse web single sign-on? An empirical investigation of OpenID. In *Proceedings of the seventh symposium on usable privacy and security*. 1–20.
[17] Manuel Urueña, Alfonso Muñoz, and David Larrabeiti. 2014. Analysis of privacy vulnerabilities in single sign-on mechanisms for multimedia websites. *Multimedia Tools and Applications* 68 (2014), 159–176.
[18] W3C. 2023. Decentralized Identifiers (DIDs) v1.0. https://www.w3.org/TR/did-core/. Accessed Online: 2023-08-23.
[19] W3C. 2023. Verifiable Credentials Data Model v2.0. https://www.w3.org/TR/vc-data-model-2.0/. Accessed Online: 2023-08-22.
[20] Rui Wang, Shuo Chen, and XiaoFeng Wang. 2012. Signing me onto your accounts through facebook and google: A traffic-guided security study of commercially deployed single-sign-on web services. In *2012 IEEE Symposium on Security and Privacy*. IEEE, 365–379.
[21] Yang Xiao, Ning Zhang, Wenjing Lou, and Y Thomas Hou. 2020. A survey of distributed consensus protocols for blockchain networks. *IEEE Communications Surveys & Tutorials* 22, 2 (2020), 1432–1465.
[22] Hakan Yildiz, Christopher Ritter, Lan Thao Nguyen, Berit Frech, Maria Mora Martinez, and Axel Küpper. 2021. Connecting self-sovereign identity with federated and user-centric identities via saml integration. In *2021 IEEE Symposium on Computers and Communications (ISCC)*. IEEE, 1–7.
[23] Yuchen Zhou and David Evans. 2014. SSOScan: Automated Testing of Web Applications for Single {Sign-On} Vulnerabilities. In *23rd USENIX Security Symposium (USENIX Security 14)*. 495–510.