Enhancing Data Exchange with Zero-Knowledge Credentials

Capstone Team: Victor Carolino, Michael Kaufman, Elizabeth Pursell, Daniella Stokic
Faculty Advisor: Brett Duersch **Special Thanks To:** Hank Korth **Sponsor:** Oracle

LEHIGH UNIVERSITY
CSB / CSE Capstone Project 2024

1 Abstract

In a world that relies heavily on handing over personal information for verification, **a more secure and efficient method of providing such data is required**. The verifiable credential data model aims to address this by utilizing modern cryptography and distributed ledgers to share specific claims without exposing unrelated personal information.

Decentralized Identifiers (DID): a self-owned, unique identifier that allows individuals to control their information without relying on a central authority.

Verifiable Credential (VC): a digital document that proves claims about an individual issued by a trusted authority.

Verifiable Data Registry (VDR): a decentralized, persistent system where non-personally identifiable information can be anchored and self-certified. Often distributed ledger technologies like blockchains are used.

2 Motivation

Within many sectors, especially healthcare, personal information gets overshared and leaked due to centralized data management. Self-sovereign identity technology greatly benefits both patients and care providers as shared information is consistent, trustworthy, and highly secure.

3 Solution

Extending the Verifiable Data Registry

A significant aspect of this project was adopting a more privacy-preserving VC format: AnonCreds. Utilizing Zero-Knowledge Proofs, AnonCreds offer additional benefits like selective disclosure and predicate proofs. We extended the provided chaincode (smart contract) to support reading and writing new data types needed for AnonCreds to the Oracle Blockchain Platform.

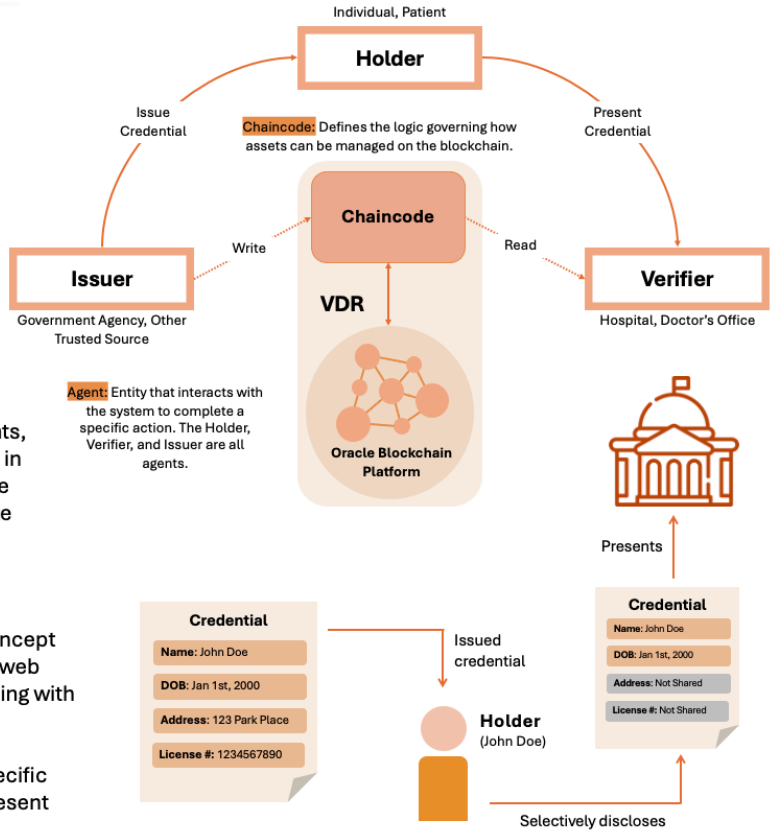
Connecting the Agents

Just like we have lawyers in the legal system, we need software, called agents, to interact with SSI systems. Agents manage the safe storage of credentials in local wallets and the handling of encrypted peer-to-peer data exchange. The team connected and tested the agents with the new endpoints in the remote VDR using TypeScript and Jest, with Postman preliminary.

Enhancing User Experience

Even with agent software handling most of the complexities, SSI is a new concept for many and not the most intuitive. That is why the team developed Next.js web frontends for role in the system (issuer, holder, and verifier) to make interfacing with the agent easy for end-users.

For the issuer, we created the ability to see incoming requests and issue specific credentials. For the holder we made it so you can request, view and then present the credentials. For the verifier, you can process presentations and verify authenticity.



4 Results & Conclusion

Despite the challenges posed by restructuring the codebase and a slight shift in project direction from spring semester, the team successfully delivered a **proof-of-concept verifiable credential system** featuring chaincode logic, issuer, holder, and verifier agents, as well as an intuitive interface for seamless interaction.

- Created an interface used to interact with Verifiable Credentials
- Transitioned from Hyperledger Aries to OpenWalletFoundation Credo
- Fully implemented chaincode for handling AnonCreds credentials
- Ensured encompassing coverage with unit and end-to-end testing

Future Enhancements

Develop a mobile interface for the holder, verifier, and issuer

Implement credential revocation so credentials aren't permanent

Expand functionality to apply to more business use cases