Hardware Assisted and Blockchain Empowered Security Lab

CENTER FOR CYBERSECURITY

FONDAZIONE BRUNO KESSLER

Improve Wallet Interoperability and Federation in Blockchain-Based User-Centric Authentication for Healthcare

Biagio Boi, Franco Cirillo, **Marco De Santis**, Christian Esposito

University of Salerno

mdesantis@unisa.it

2nd International Workshop on Trends in Digital Identity (TDI 2024)

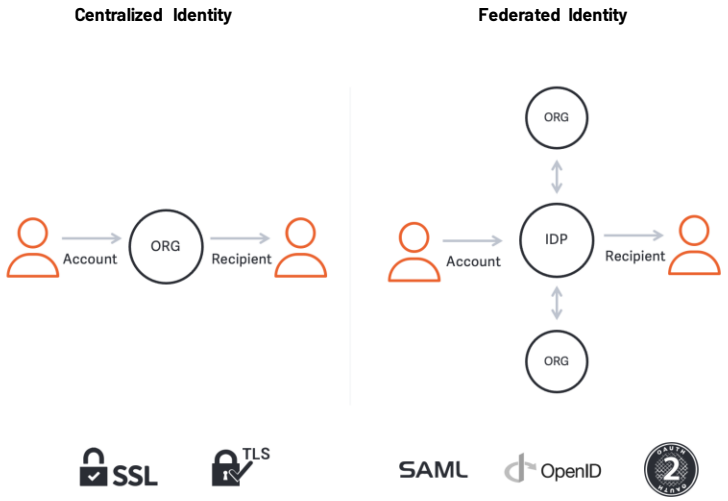
Introduction

Authentication has always been a critical component when considering to protect a system.

Recently, these mechanisms have changed frequently to increase security and privacy for users.

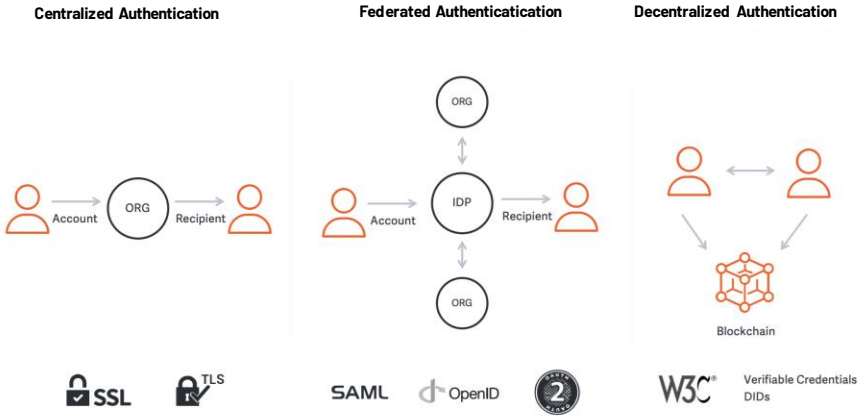
Authentication Server (AS) can be seen as a single point of failure, compromising security and availability of users data.

A centralised AS can profile the users' habits when surfing the web, causing privacy violations.



Centralized Identity vs. Federated Identity

Introduction



Centralized vs. Federated vs. Decentralized Authentication

The main advantages of using federated authentication are related to the way in which user data is managed, which is particularly relevant in the medical context, where stored data is particularly sensitive.

Despite the advantages of this approach, which does not require credentials to be stored on each service server, some privacy and security shortcomings must be taken into account.

Decentralized authentication emerges as a promising solution to address security and privacy challenges in digital authentication.

Novel Authentication Methods and Hardware Devices:

Password-less Authentication

Passwordless authentication methods, such as biometric authentication using fingerprint scanners, offer users a more secure and convenient way to access their accounts.

Proof-of-possession mechanisms

Hardware devices, such as smart cards, can be used as proof-of-possession mechanisms, adding an additional layer of security to the authentication process.

Authentication in Medical Domain

Need for Robust Identity Management Solutions

In the medical domain, there is a crucial need for robust identity management solutions to address the growing concerns of data breaches and the limitations of centralized identity and claim management systems.

Data Breaches

The medical domain is a prime target for data breaches due to the sensitive nature of patient information. These breaches can result in compromised patient data, financial loss, and reputational damage to healthcare organizations.

Limitations of Centralized Identity Systems

Centralized identity and claim management systems have limitations in terms of scalability, privacy, and security. These systems often rely on a single point of failure, making them vulnerable to attacks and unauthorized access.

Challenges for Healthcare Professionals

Adoption of New Technologies

Health care providers face challenges in adopting new technologies such as sovereign identity wallets (SSIs). Unfamiliarity with these tools and the need for education and training can hinder effective and secure use.

Ensuring Effective and Secure Usage

Education and training are critical to ensure that health care professionals understand the benefits and risks associated with new authentication methods and verifiable credentials. This knowledge is essential for their effective and secure use in medical settings.

Proposed Solution:
Credentials Restore and User Differentiation

Leveraging self-sovereign identity (SSI)

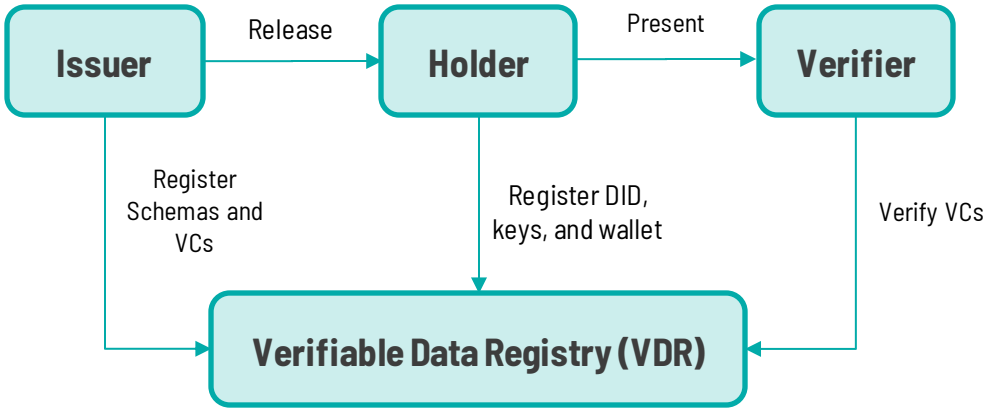
Using SSI solutions, we can provide a secure and decentralized way to restore credentials and differentiate users in the medical field.

Benefits of the proposed solution

- **Privacy and Security**: SSI provides individuals with greater control over their personal health information, ensuring that it is only shared with authorized parties and reducing the risk of data breaches.
- **Interoperability**: SSI enables seamless sharing of health information between different healthcare providers and systems, improving care coordination and patient outcomes.
- **Patient Empowerment**: SSI allows individuals to actively participate in their own healthcare by granting them access to their medical records and enabling them to make informed decisions about their treatment.

SSI: a novel decentralized apporach

Self-Sovereign Identity (SSI) aims to remove Authentication Servers (AS) from authentication procedures by leveraging the concept of Verifiable Credentials. In such a schema, each entity holds a VC, which is used for the verification and is identified by a Decentralized Identifier (DID).



Different Levels of Responsibility in the Medical Context

In the medical context, there are different levels of responsibility that require tailored authentication measures based on the impact of compromised identities. These levels include:

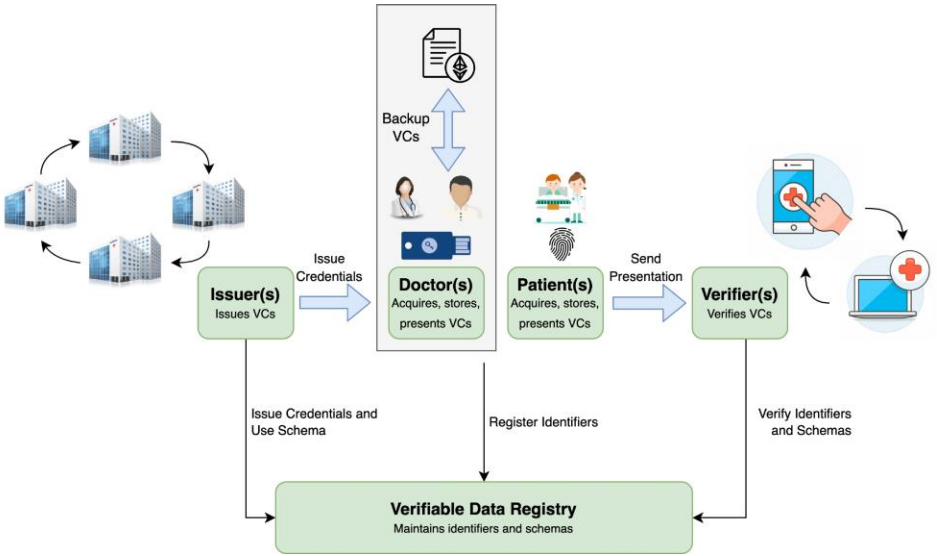
1. **Medical Professionals**: Healthcare providers, doctors, nurses, and other medical professionals who have access to sensitive patient information and are responsible for providing accurate diagnoses and treatments.

2. **Administrative Staff**: Personnel responsible for managing patient records, scheduling appointments, and handling billing and insurance information.

3. **Patients**: Individuals who have the right to access their own medical records and make informed decisions about their healthcare.

It is crucial to implement authentication methods that ensure the security and privacy of patient information while allowing authorized individuals to perform their respective roles effectively.

Proposed Architecture

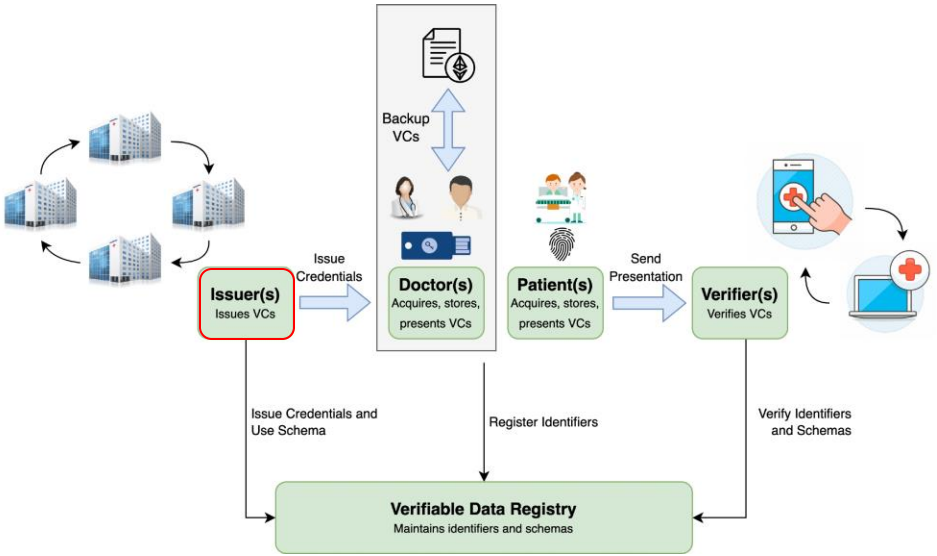


Our system operates within the framework of the trust triangle, which comprises three key entities:

- Issuers
- Holders
- Verifiers

Each entity plays a distinct role in the authentication process, contributing to the overall security and reliability of the system.

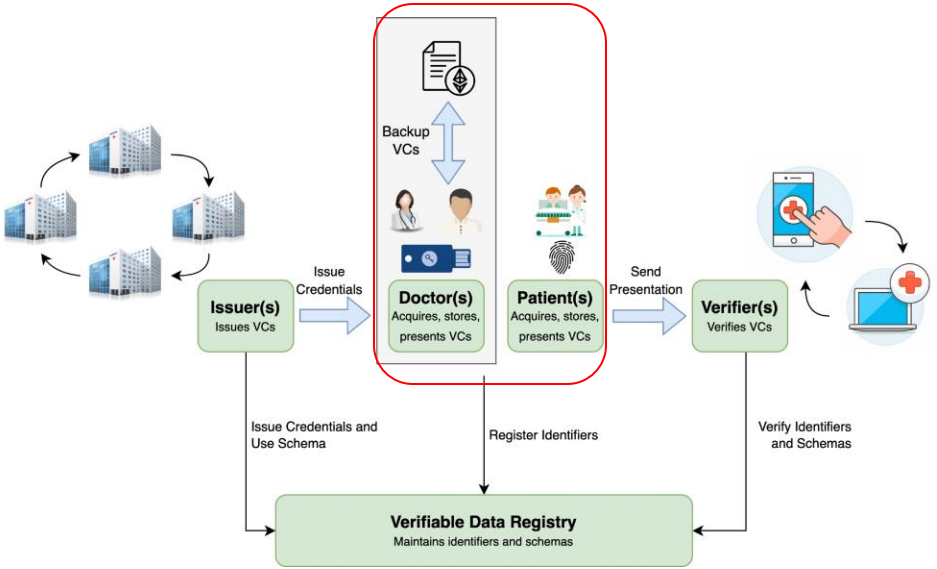
Proposed Architecture



Our system operates within the framework of the trust triangle, which comprises three key entities:

Issuers : Hospitals serve as autonomous issuers within our system, responsible for issuing Verifiable Credentials (VCs) to patients and healthcare providers. By leveraging decentralized identifiers (DIDs) and cryptographic keys, hospitals can securely issue and manage credentials, ensuring the integrity and authenticity of the information

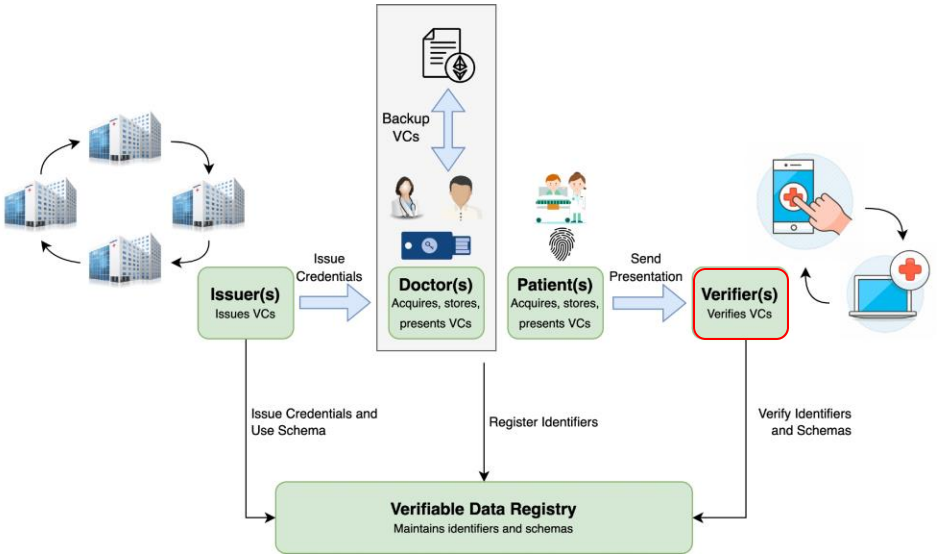
Proposed Architecture



Our system operates within the framework of the trust triangle, which comprises three key entities:

Holders : Credential holders are doctors and patients who hold a cryptographic wallet containing credentials issued by hospitals. Doctors use a desktop wallet, while patients use a mobile wallet to access health care services.

Proposed Architecture

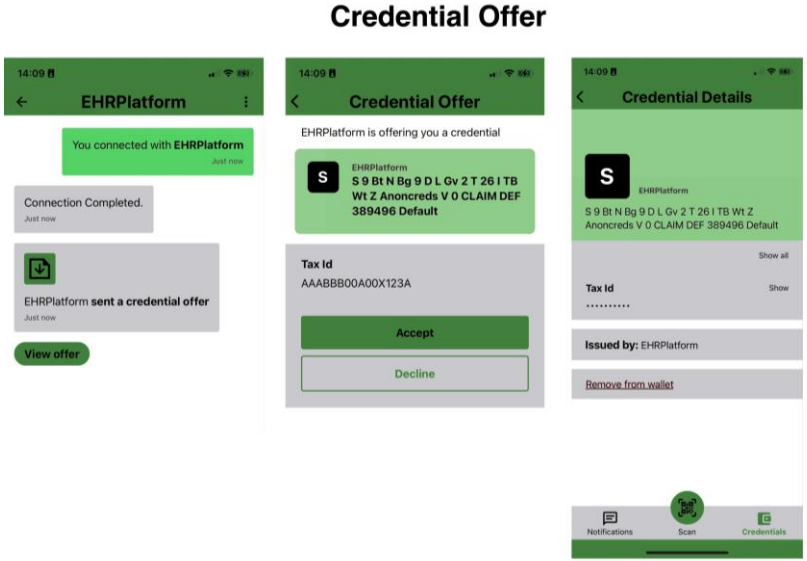


Our system operates within the framework of the trust triangle, which comprises three key entities:

Verifiers: Healthcare platforms and systems serve as a verifier, responsible for verifying credentials submitted by physicians and patients to access health care services ensure that only authorized individuals can access sensitive information, thereby safeguarding patient privacy and confidentiality.

Proposed Architecture: Patients Wallet

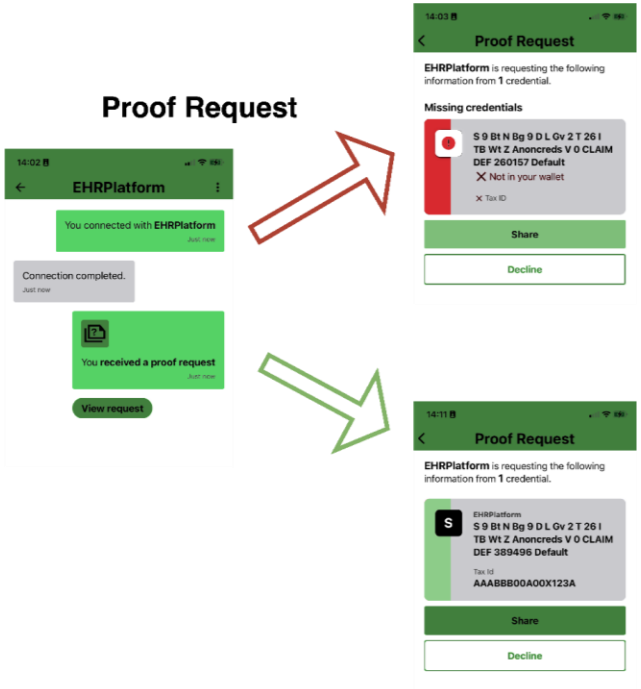
Patient(s)



Through a mobile application, pasients can interact with the ledger and manage their credentials which is based on the Credo.js platform.

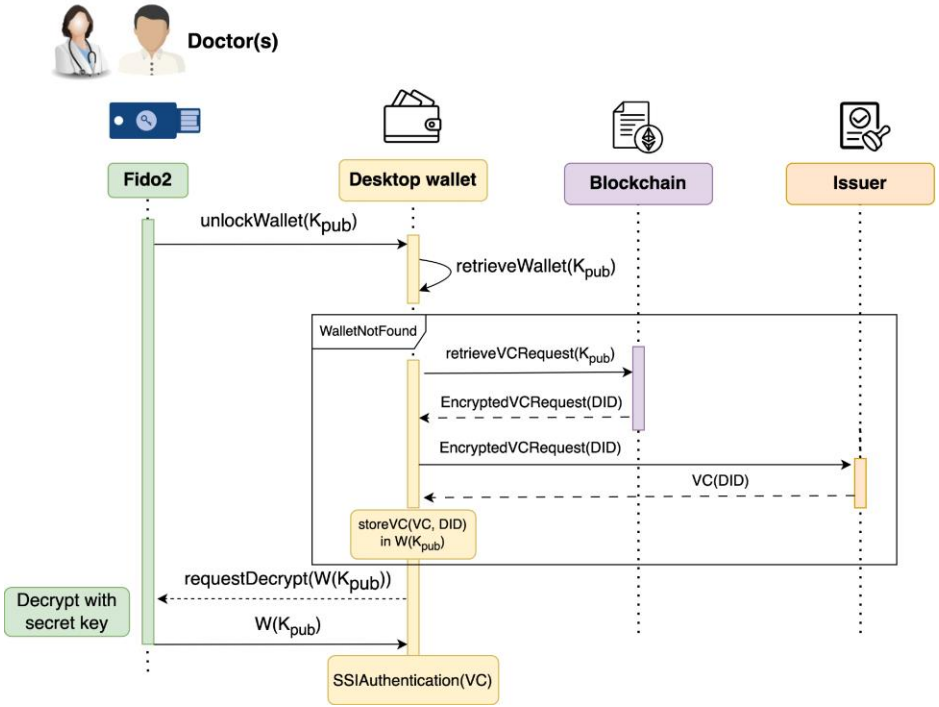
The Patients Wallet allows users to accept proposals for new Verifiable Credentials (VCs) by scanning a QR code to establish the connection and then accepting or rejecting proposals for new VCs. Once the credentials are stored in the wallet, the user can use them for authentication on the platform

Proposed Architecture: Patients Wallet



The Patients Wallet also allows users to access services by initiating connections with the service provider and waiting for Verifiable Presentations (VP) requests.

Proposed Architecture: Doctor Wallet



The system uses a desktop wallet approach, leveraging Veramo's SDK to manage multiple wallets in a single device through encryption.

Each doctor holds a single key pair associated with a FIDO2 device, which is used for authentication.

The desktop wallet is used to generate a decentralized public identifier (DID) for the wallet and encrypt the associated credentials.

The system also uses a Trusted Platform Module (TPM) to protect the key pairs used in the SSI ecosystem.

Verifiable Credentials (VCs) Backup

In our project we propose a smart contract-based system that can provide backup and recovery capabilities for verifiable credentials (VC) through the use of a Trusted Platform Module (TPM).

```
contract CredentialManager {
address public owner;
mapping(address => bool) public isIssuer;
mapping(address => bytes32) public addressToPublicKey;
mapping(address => string) public addressToEncryptedUrl;

modifier onlyOwnerOrIssuer() {
        require(msg.sender == owner || isIssuer[msg.sender], "Only
issuer can call this function");
_;
}

event AddressAdded(address indexed ethAddress, bytes32 publicKey,
        address indexed addedBy);
event AddressRemoved(address indexed ethAddress, address indexed re
event CredentialMapped(address indexed ethAddress,
        string encryptedUrl, address indexed mappedBy);

AddressList public addressList;

constructor() {
        owner = msg.sender;
        isIssuer[msg.sender] = true;
        addressList = new AddressList();
}
```

We defines a mapping between a public address and a string called **addressToEncryptedUrl**.

This mapping allows the issuer to associate a credential release request with a given user wallet associated with the FIDO2 key by exploiting the key generation function.

Verifiable Credentials (VCs) Backup

To prevent attacks, we decided to encrypt the credential release URL with the $Kpub$ key stored in the key so that only the true owner can decrypt the content using the $Kpriv$ key.

When the wallet retrieves the credential request, it can proceed with the retrieval by automatically receiving the credentials from the issuer.

```
function addAddress(address ethAddress, bytes32 publicKey)
 public onlyOwnerOrIssuer {
        addressToPublicKey[ethAddress] = publicKey;
        addressList.addAddress(ethAddress);
        emit AddressAdded(ethAddress, publicKey, msg.sender);
}

function mapCredential(address ethAddress, string memory encryptedUrl)
public onlyOwnerOrIssuer {
        require(addressToPublicKey[ethAddress] != 0, "Address not
authorized");
        addressToEncryptedUrl[ethAddress] = encryptedUrl;
        emit CredentialMapped(ethAddress, encryptedUrl, msg.sender);
}

function retrieveCredential(address ethAddress)
public view onlyOwnerOrIssuer returns (string memory) {
        require(bytes(addressToEncryptedUrl[ethAddress]).length > 0, "No
credential mapped for this address");
        return addressToEncryptedUrl[ethAddress];
}


function removeAddress(address ethAddress)
public onlyOwnerOrIssuer {
        delete addressToPublicKey[ethAddress];
        delete addressToEncryptedUrl[ethAddress];
        emit AddressRemoved(ethAddress, msg.sender);
}

function updateIssuer(address issuer, bool isIssuerAllowed)
public onlyOwnerOrIssuer {
        isIssuer[issuer] = isIssuerAllowed;
}
```

Conclusion

Implementing an SSI-based system requires careful consideration to prevent attacks on Verifiable Credential (VC) exchanges and ensure overall system security.

The next step involves evaluating physician feedback and assessing the effectiveness of the proposed mechanism in real-world settings.

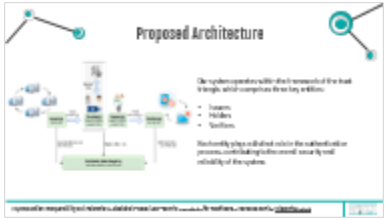
This process will allow for adapting the system to the specific needs of the medical context and ensuring optimal long-term implementation

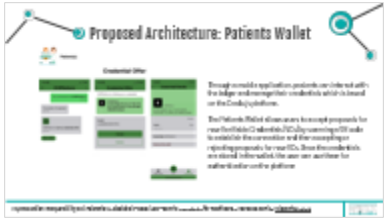
Thank you!

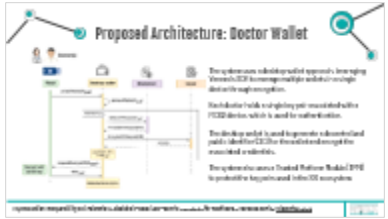
Questions?

Biagio Boi, Franco Cirillo, **Marco De Santis**, Christian Esposito

Improve Wallet Interoperability and Federation in Blockchain-Based User-Centric Authentication for Healthcare







Hardware Assisted and Blockchain Empowered Security Lab