Inji:
Inside the Credentials Wallet

Harini Sampathkumar | Product Owner, MOSIP

Inji:
Decentralised Wallet for Credentials

Provisioning:

- VC Issuance via eSignet as auth layer
- Cryptographic holder binding from Issuer to Wallet

Trust Enabled Presentation:

- Cryptographic ephemeral key pair generation
- Encrypted data transfer via BLE
- Presence assurance: Decentralised face verification

Credentials
and Proof Supported

VC Format:
- W3C JSON-LD

Proof Types:
- RSA
- ED25519

Inji:
Extensibility & Modularity

Modules that can be integrated to existing infrastructure

- Tuvali
  - Leverages Bluetooth Low Energy (BLE)
  - Offline peer to peer VC transfer from Wallet to Verifier
  - Secure transfer of data through cryptographic key exchange

- Secure-keystore (Android)
  - Hardware backed secure storage for creating and storing key-pairs
  - Supports RSA based Key pair and symmetric keys
  - HMAC based verification and tamper protection of data

Road Ahead:

Library for:

- VCIssuance
- VCVerification

Road Ahead

Road Ahead – Digital Wallet (INJI)

Long Term

- User profiles
- Query and predicates
- Cloud Wallet
- USSD Channel
- Attestation Module

Plans for 2024

- Data backup
- Selective disclosure using SDJWT
- Revocation support
- Online sharing
- Update to OpenID4VCI drafts

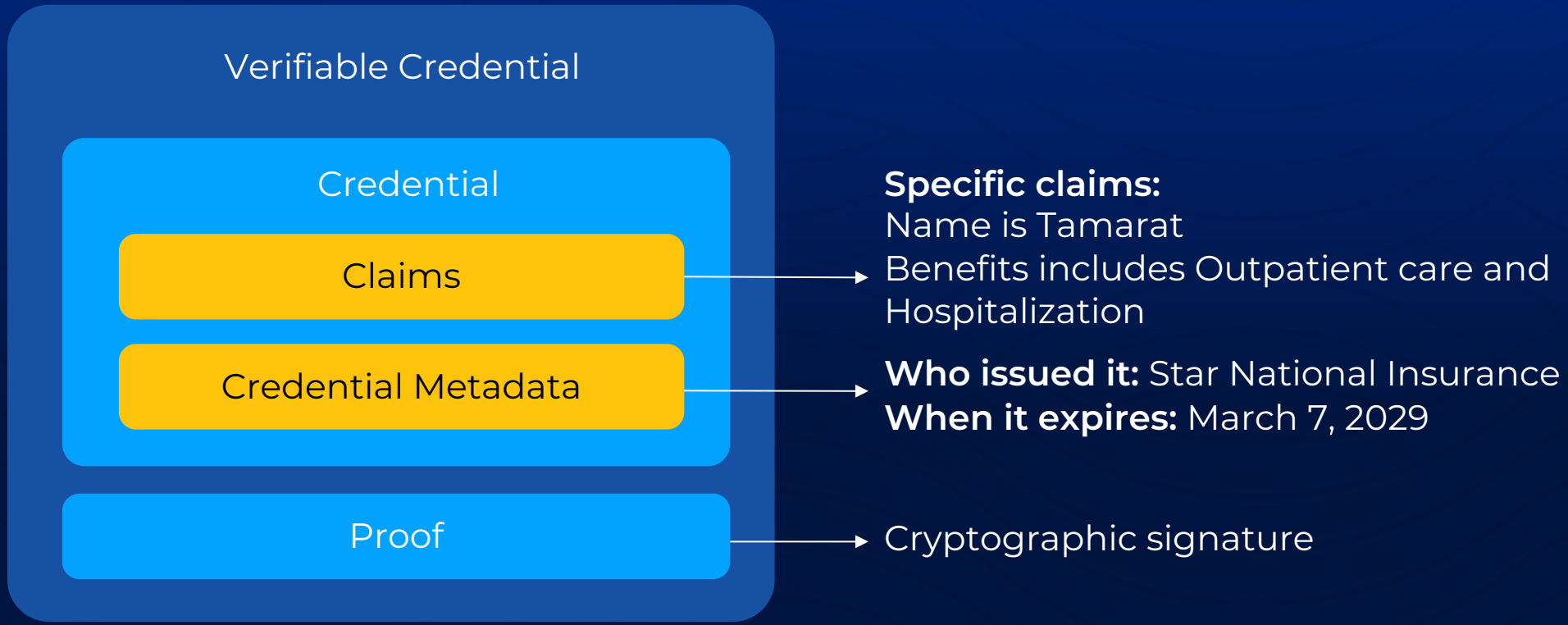
Exploring Credential Formats

A Comprehensive Overview

Vishwanath Vaidyanathan | Principal Architect, MOSIP

What is
a Credential Format?

Encapsulates the Claims and Proof in a standard manner for the holders and verifiers to parse and understand



Essential Features

Understand Features to Demystify Credential Formats

Encoding
Schemes

- Claims and metadata are structured inside a credential using encoding schemes
- Some formats includes proof into the encoding

- Commonly used encoding schemes:

JSON

- JavaScript Object Notation
- Lightweight data-interchange format
- Machine readable
- Human friendly

JSON LD

- JavaScript Object Notation for Linked Data
- All advantages of JSON
- Semantically markedup data
- Machine understandable

CBOR

- Concise Binary Object Representation
- Based on JSON data model
- Binary based - smaller size
- Not human readable
- Extensibile binary format

Signing Algorithms

- Signatures makes the Digital Credential **tamper evident** and **enables trust**
- National regulation agencies analyses and recommends secure **cryptographic** algorithms
- Hardware support should be considered for regulated and **high-security** use cases to prevent key duplication and theft
- With the computing power of quantum computers advancing, **post-quantum security** aspects should also be considered

- Widely used signature algorithms
  - ECDSA
  - EdDSA
  - RSA
  - BBS+
  - CL

Key Management

- Keys used for signature generation should be **securely** managed
- Key rotation policies should be place to reduces the risk of using **compromised keys**
- Key revocation process allows a **verifier** to be aware that credentials should not be trusted if they were signed using a compromised key
- Public keys for verification should be **resolvable** in a **trusted** manner

- Widely used key resolution methods
  - .well-known/jwt-issuer
  - raw public keys
  - did:web
  - did:jwk

Holder
Binding

Allows a holder to prove that they are legally in possession of a VC

Cryptographic binding

- Credential contains a public key or a reference to a public key that corresponds to the private key controlled by the Holder

Claim based binding

- Binded to claims like name and date of birth, which can be proved by presenting another Verifiable Credential
- Allows long-term, cross-device use of a Credential as it does not depend on cryptographic key material stored on a certain device

Biometrics based binding

- Allows verification by demonstrating a certain biometric trait, such as fingerprint or face

No binding

- Helps with use cases like coupons, where binding is not a requirement

Selective Disclosure

Specific transactions needs only specific claims to be presented to a verifier, but generally credentials are one unit containing multiple claims together

Selective disclosure allows a holder to present a subset of the attributes of the credential issued by the issuer.

Credential schema can also be improved to include abstract claims, such as ageOver

Uncorrelatability

- Blind signatures - algorithm like BBS+ or CL signature
- Zero-Knowledge Proofs

Predicates

- Further decreases the amount of information shared by checking a value against a certain condition, resulting in true or false

Compound proofs

- Proving values are same among multiple credentials from different issuers

Popular
Credential Formats

Format / Profile	Encoding Scheme	Signing Algorithm	Hardware Support	Selective Disclosure Support	Predicates Support	Unlinkability
W3C JWT VC	JSON	ECDSA, EdDSA, RSA	Yes	No	No	No
W3C JSON-LD VC	JSON LD	ECDSA, EdDSA, RSA	Yes	No	No	No
IETF SD-JWT	JSON	ECDSA, EdDSA, RSA	Yes	Yes	No	No
W3C JSON-LD/BBS+ VC	JSON LD	BBS+	No	Yes	Yes	Yes
ISO MDOC	CBOR	ECDSA, EdDSA, RSA	Yes	Yes	No	No
Anon Creds	JSON	CL	No	Yes	Yes	Yes

Our Focus

- Our focus have been on machine understandable credentials and privacy preserving credentials
- We are also interested in Compact credentials – CBOR encoded QR Codes

- We currently support:
  - Credential Format: W3C JSON-LD VC
  - Signing Algorithm: RSA
  - Key Resolution: did:web, did:jwk
  - Holder Binding: All types

- Support for EdDSA based proof is in progress
- Road Ahead:
  - Selective disclosure using IETF SD-JWT
  - ECDSA crypto enhancement
  - Revocation support

MOSIP

MOSIP Homepage: www.mosip.io

MOSIP Source Code: github.com/mosip

MOSIP Documentation: docs.mosip.io

MOSIP Community: community.mosip.io