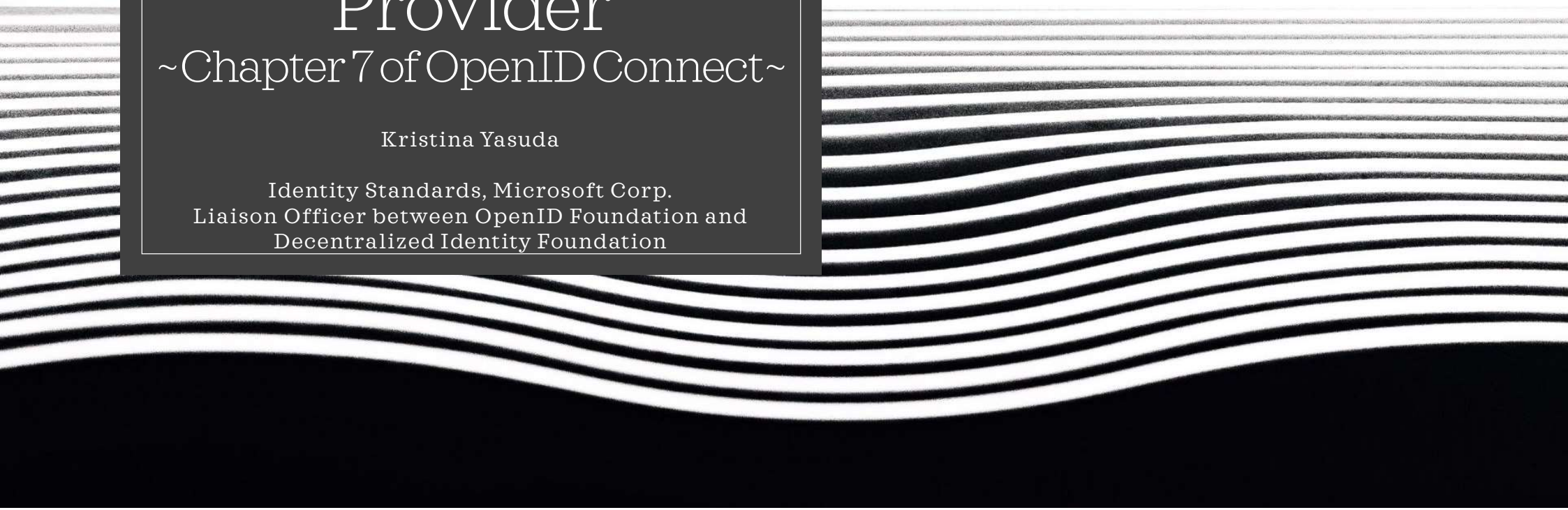
Self-Issued OpenID Provider

~Chapter 7 of OpenID Connect~

Kristina Yasuda

Identity Standards, Microsoft Corp.
Liaison Officer between OpenID Foundation and
Decentralized Identity Foundation

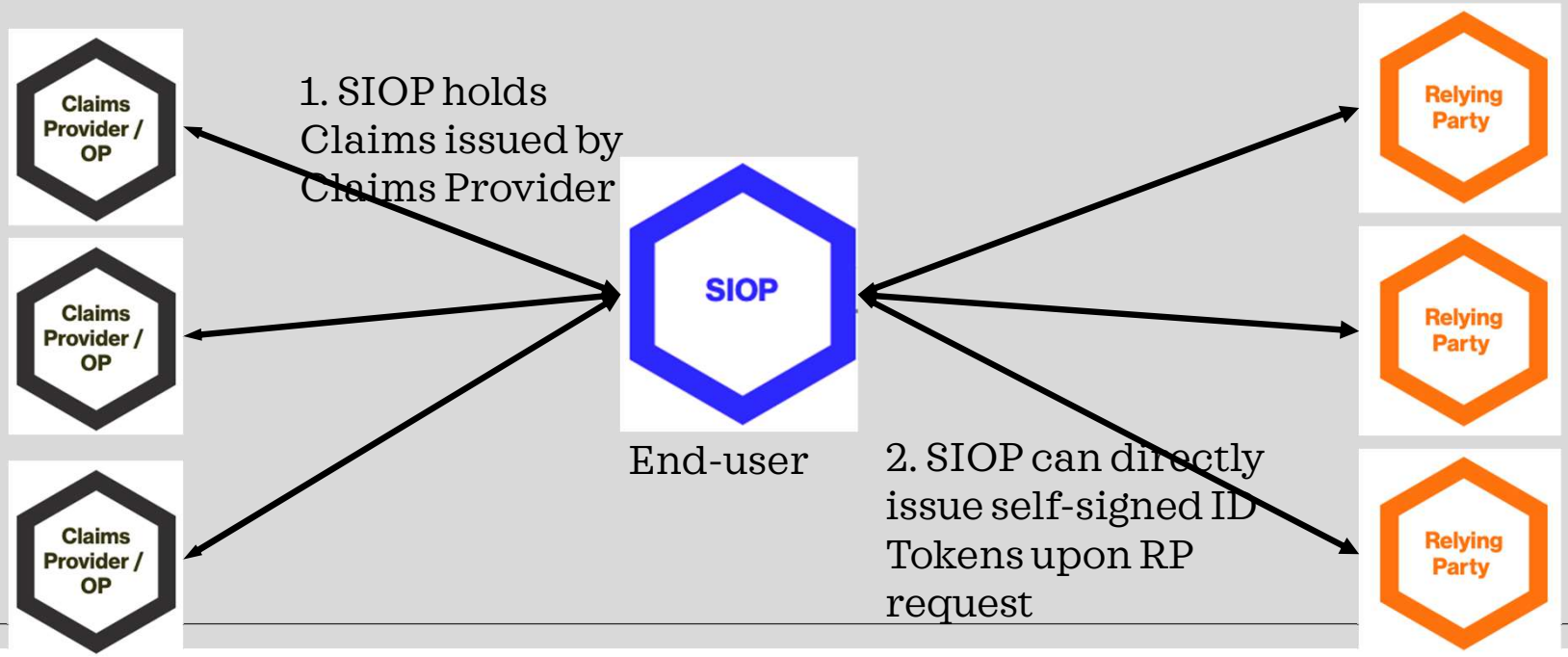


Caveat: WIP. Not everything reflects OIDF Connect WG consensus.

- DIF-OIDF liaison
- Requirements
- Scopes
- Drafts

What is Self-Issued OpenID Provider (SIOP) ?

- Self-Issued OpenID Providers are personal OpenID Providers that issue self-signed ID Tokens, enabling portability of the identities among providers.
- User holds its own OpenID Provider(OP) <> No Central OP
- Chapter 7 of OpenID Connect



Self-Issued OpenID Connect Provider DID Profile v0.1

- Working Group Draft at Decentralized Identity Foundation
- DIDAuthn flavor to use OIDC with Decentralized Identifiers (DIDs)
- Limitations of a profile
  - Additional scope
  - Additional crypto
  - Additional properties
  - Etc.

DIF-OIDF liaison

Purpose: “Provide a mechanism for the parties to work together on the exchange of best practices, community coordination, communications … in areas of mutual interest to the organizations.”

Self-Issued OpenID Provider as one of the first Collaboration items.

Work on Self-Issued OpenID Provider DID Profile moved from DIF to OIDF Connect WG.

Decentralized Identity Foundation

- **Neutral** place for community to collaborate; define **emerging specifications** on decentralized identity, demonstrate **interoperability**.
- Focus on **open source & specs**
- Nonprofit, founded in 2017; 150+ members, Part of Linux foundation
- 87 Github repos, 7 technical groups, open groups …

https://identity.foundation/

SIOP Requirements (List)

openid/connect/src/master/SIOP/siop-requirements.md

A. SIOP request
B. SIOP response
C. Key recovery and key rotation
D. Trust model between RP and SIOP
E. Issuance of the claims
F. Privacy protection
G. Claims binding
H. Various OpenID providers deployment architectures
I. Use-case specific requirements

SIOP Requirements draft (Excerpt)

B. SIOP response

4. SIOP should be able to return Verifiable Credentials and Verifiable Presentations in the response

C. Derivation of Key information (cryptography itself is out of scope)

5. Key information should be derived either by using Decentralized Identifiers resolved into DID documents, or sub_jwks with URNs (-> deep-dive)

D. Trust model between RP and SIOP (Invocation, Discovery, Registration)
+ accounting for a special use-case where RP and SIOP are on the same device

E. Issuance of the claims (SIOP - Claims Provider)

9. SIOP providers can be registered with the Claims provider (Unique to SIOP)

Scopes

1. Enabling ‘portable’ subject identifiers between providers - Define how to use techniques such as asymmetric cryptography like Decentralized Identifiers to create subject identifiers that are not intrinsically bound to a particular OP and hence can be ported between providers

2. Solving for provider discovery and registration - Define how does an RP come to have a relationship and understand capabilities of an OP, and what role the user plays in this selection/discovery process. (+ NASCAR problem)

3. RP - OP co-location on the same device – Define how to deal with the unique requirements that are brought about when the OP is communicating with the RP on the same device (e.g in the form of a PWA or Native App), rather than a traditional Authorization server.

4. Credential Issuance support - Issuing credentials from OpenID Connect flows.

5. Credential Presentation support - Presenting credentials in OpenID Connect flows.

Drafts

Self-Issued OpenID Provider V2, draft 01

Covers three scopes

1. Cryptographically verifiable identifiers (DIDs),
   - Introducing a layer of indirection by allowing `sub` claim to be a URN, so that it can be both jwk thumbprint and DIDs
   - Updating verification methods when DIDs are included in `sub`?
   - Additional cryptography mechanisms (ES256K/EdDSA)

2. Discovery and Registration
   - Defined the ability to pass the registration information both by value and by reference
   - Another idea exists to introduce a mechanism where upon issuance, issuer includes data in a credential that wallet can use to register itself at the endpoint and that RP can use to discover the wallet. Requires agreement between the Issuer and RP on details.

3. Credential Presentation Support
   - Introducing 'vp' claim in response (verifiable presentation as in W3C VC-data-model specification)

Drafts

OpenID Connect Credential Provider

Scope is Credential Issuance (Claims Binding)

- Introducing a mechanism to bind id_token to the Holder(Client) who requested it, to make user assertion suitable as a credential
- In the request, includes key material to which the Holder(Client) is requesting the credential to be bound to and the key responsible for signing the request object.
  - How client proves control over the keys? Signed request? DPoP?
- Optional DID support
- Optional JSON-LD support
- Authorization code flow

Discussions during OIDC AB/Connect WG calls:

- Weekly Pacific time-zone calls and
- Bi-weekly Atlantic time-zone calls

+ Bitbucket issues, drafts ☺