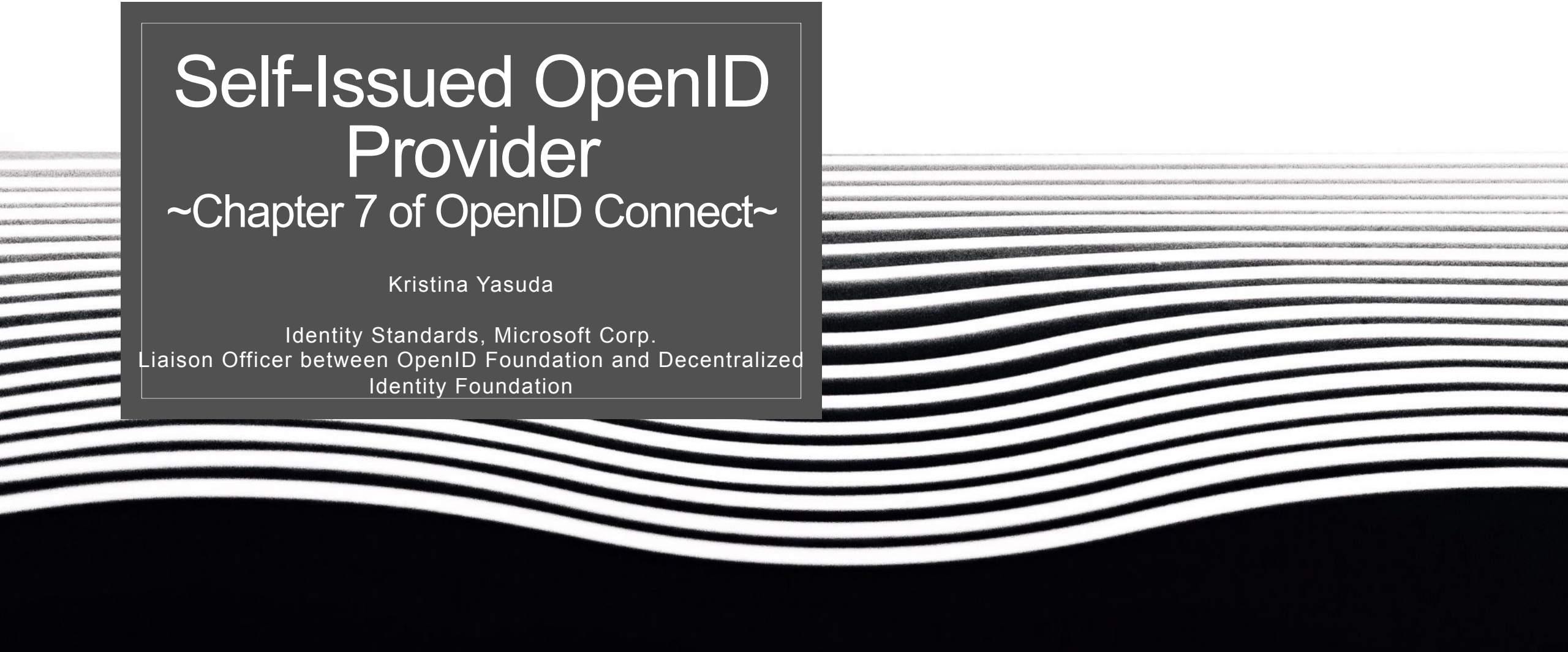
Self-Issued OpenID Provider

~Chapter 7 of OpenID Connect~

Kristina Yasuda

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

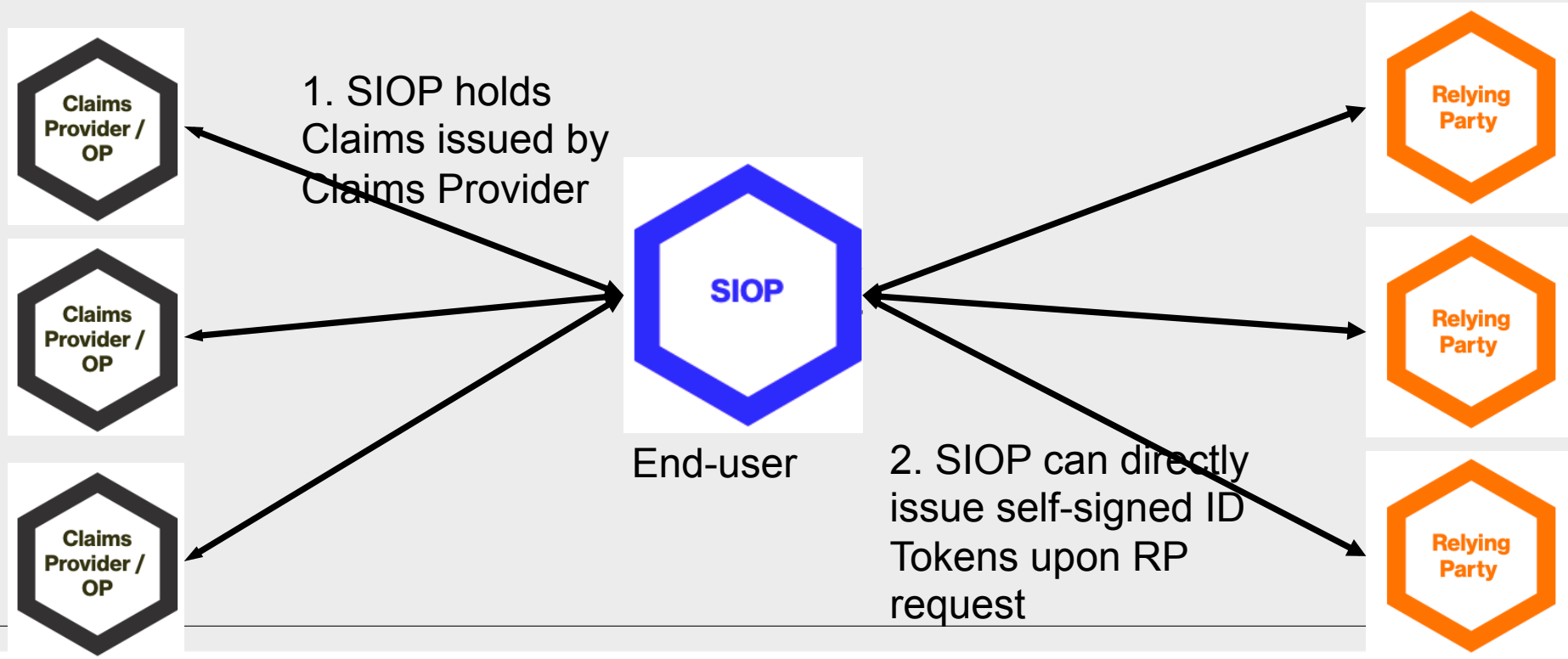


A session with a lot of open questions

1. What is Self-Issued OpenID Provider (SIOP) ?
2. SIOP Requirements (draft)
3. Initial discussion points deep-dive

1. 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



2. SIOP Requirements draft (1/4)

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

2. SIOP Requirements draft (2/4)

A. SIOP request

1. OpenID Provider's capability to issue self-issued responses is an extension of the core OpenID Connect protocol => redirect_uri
2. SIOP can be used both for logins and for transmitting identity characteristics.
3. SIOP should support best practices of flow types.

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)

2. SIOP Requirements draft (3/4)

D. Trust model between RP and SIOP (accounting for a special use-case where RP and SIOP are on the same device?)

6. SIOP must be able to advertise that it is a SIOP-enabled OP => Invocation (-> deep-dive)
7. SIOP must be able to advertise configuration information to the RP => Discovery
8. RP must be able to register with SIOP => Registration parameter

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

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

F. Privacy protection

10. RPs should understand the security/privacy posture of SIOP
11. SIOP should support pairwise, omnidirectional, and ephemeral identifiers
12. Attestations made in the past should remain valid
13. RP must be able to receive the claims when the end-user is offline without colluding with the Claims Provider

2. SIOP Requirements draft (4/4)

G. Claims Binding (relation with Aggregated and Distributed Claims Draft?) (OpenId Connect Credential Provider draft?)

H. Various OpenID providers deployment architectures (Authentication flows?)

14. Support PWA-based SIOP implementations
15. SIOP should support browser flow path, device flow path and combination of both

I. Use-case specific requirements

16. SIOP could support rich identity information sharing with RP (optional)
17. SIOP should allow for selective disclosure of claims in claim sets
18. SIOP should allow offline authentication

3. Discussion points deep-dive

1. Finding the SIOP address (Issue #1199) re: NASCAR Problem

If there are several SIOP wallets on my mobile device (or in a web browser), which one gets invoked when SIOP request is received?

Currently, SIOP wallets would register custom schema openid://. However, there are certain dependencies on the OS that does not allow to choose among wallets registered under the same custom schema.

Is there a way to make this work without OS support (ideal), or should the conversation with OS vendors be initiated (hard)? One idea was to have a “capability broker“ that registers a list of SIOP wallets and the identifier methods they support (jwk thumb or did methods)

From a user experience perspective, leaving current openid:// schema mechanism could work fine – no user confusion over existence of several wallets.

3. Discussion points deep-dive

2. Conduit to Decentralized Identifiers

- “Decentralized Identifiers (DIDs) allow DID controller(end-user) to prove control over an identifier without requiring permission from any other party”
- Advertising support for DIDs?
  - Extension to `subject_types`? New parameter `identifier_type`?
- Where to best represent DIDs – key pair controlled by you?
  - Introducing indirection to `sub` claim allowing it to be a URN allowing both jwk thumbprint and DIDs
  - `iss` is self-issued.me and has to be a URL per OpenID Connect Specification
- Updating verification methods when DIDs are included in `sub`?
- Additional cryptography mechanisms required (ES256K/EdDSA)

Collaboration with Decentralized Identity Foundation (DIDAuthn WG)

Discussions during OIDC AB/Connect WG calls:

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

+ Bitbucket issues, drafts ☺