OAuth Identity and Authorization Chaining Across Domains

IETF OAuth Interim Meeting: 16 Dec 2024

Arndt Schwenkschuster (SPIRL)
Pieter Kasselman (SPIRL)
Brian Campbell (Ping)
Mike Jenkins (NSA-CSS)
Kelly Burgin (MITRE)

Agenda

1. Recap: Identity and Authorization Chaining across Trust Domains
2. Topic for Discussion - Sender constraining tokens
3. Working Group Wisdom Requests

Why Identity Chaining Across Trust Domains

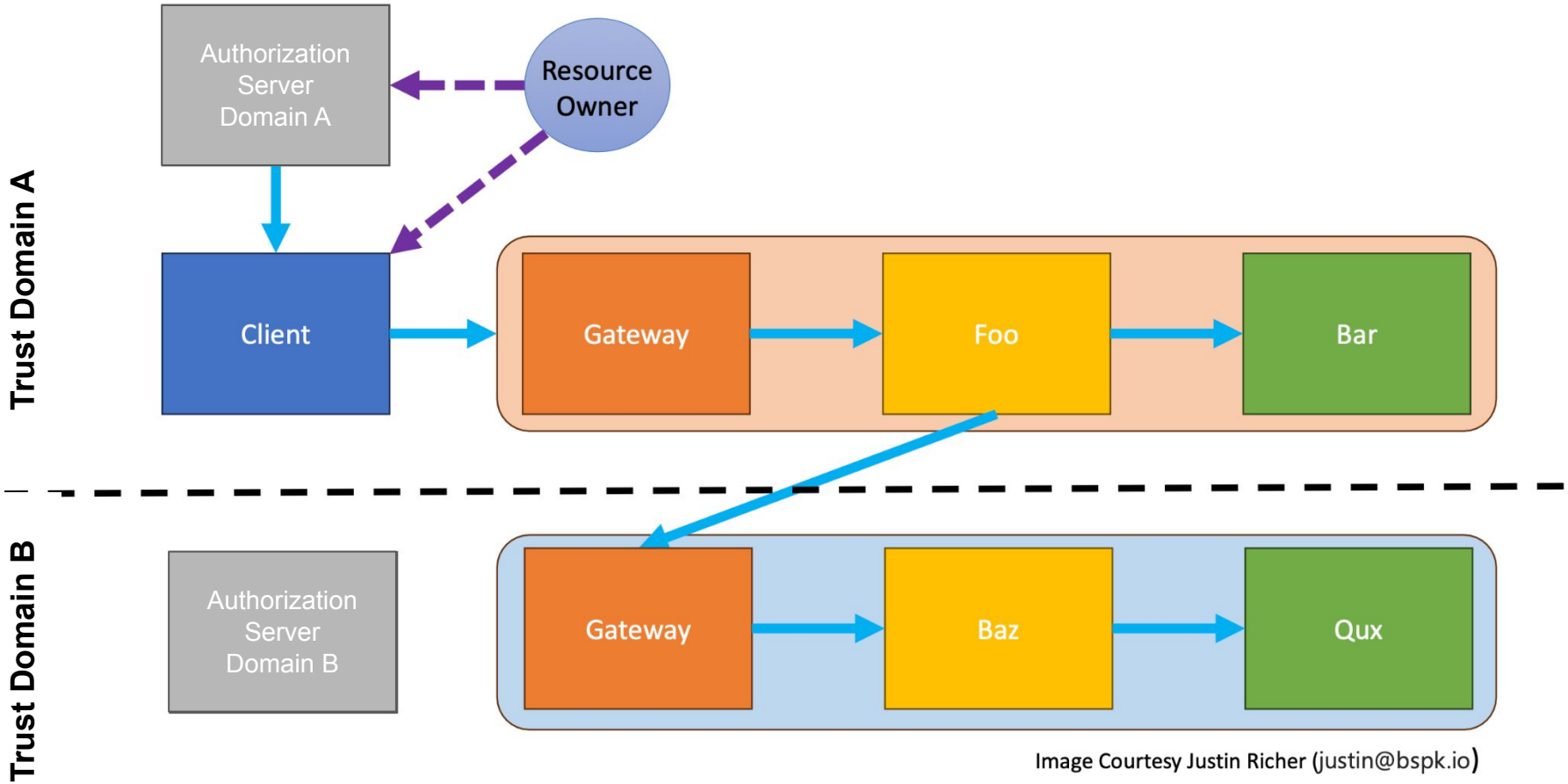


Image Courtesy Justin Richer (justin@bspk.io)

Why Identity Chaining Across Trust Domains

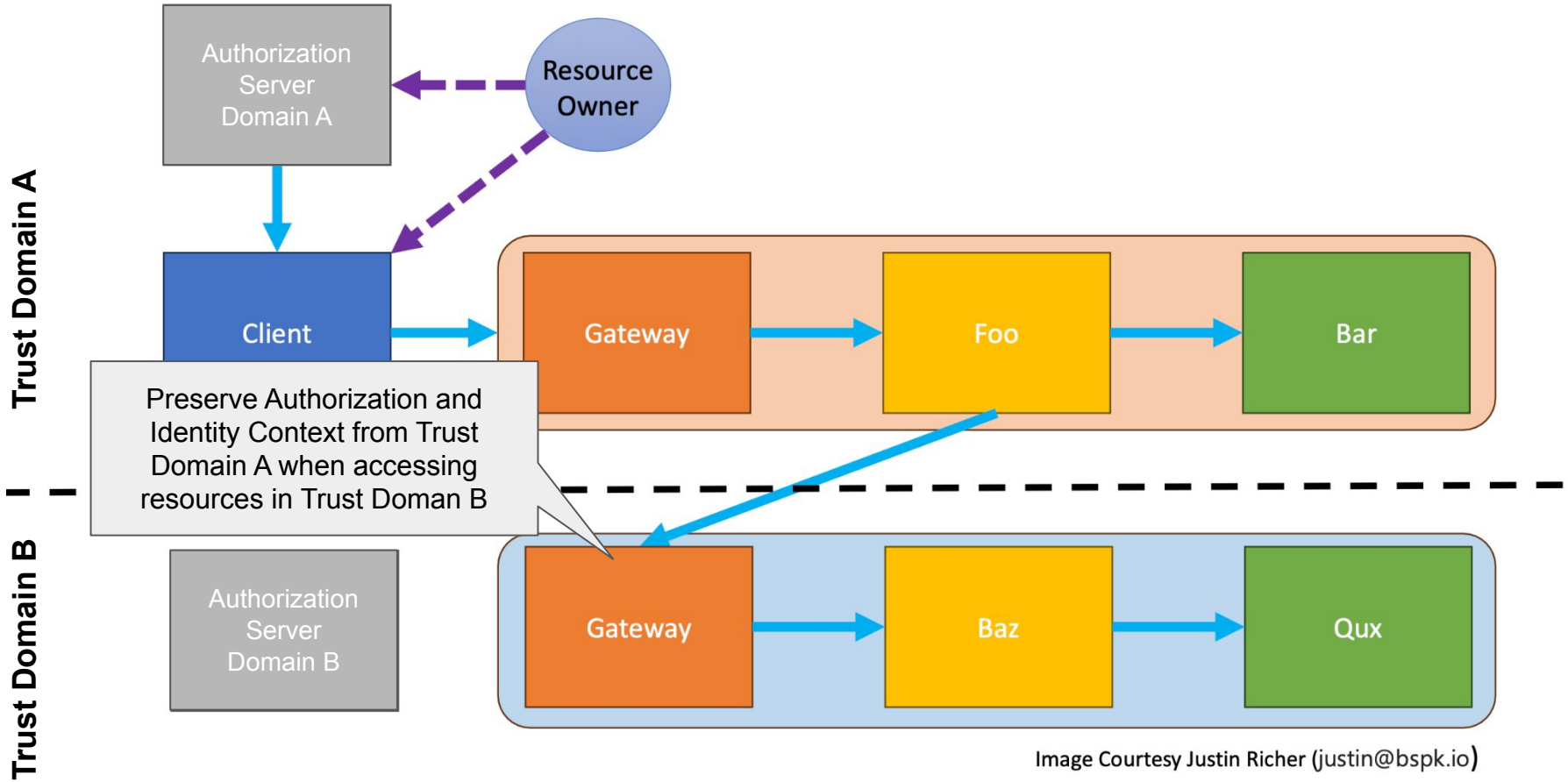
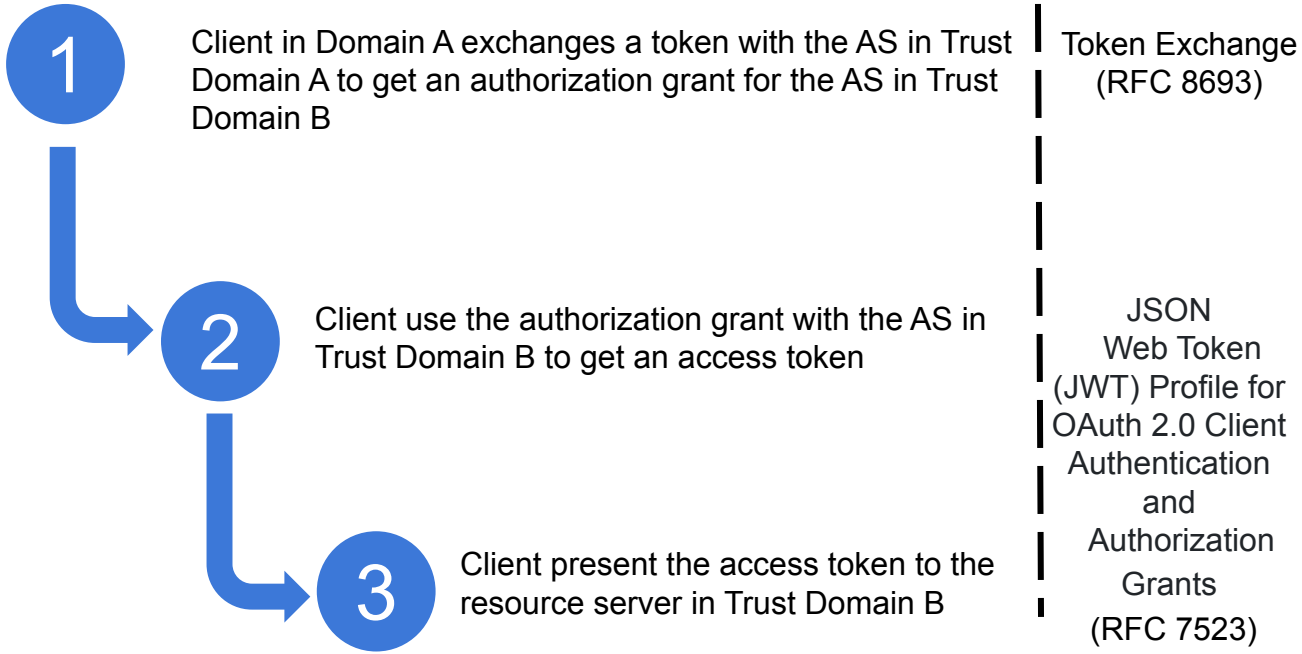
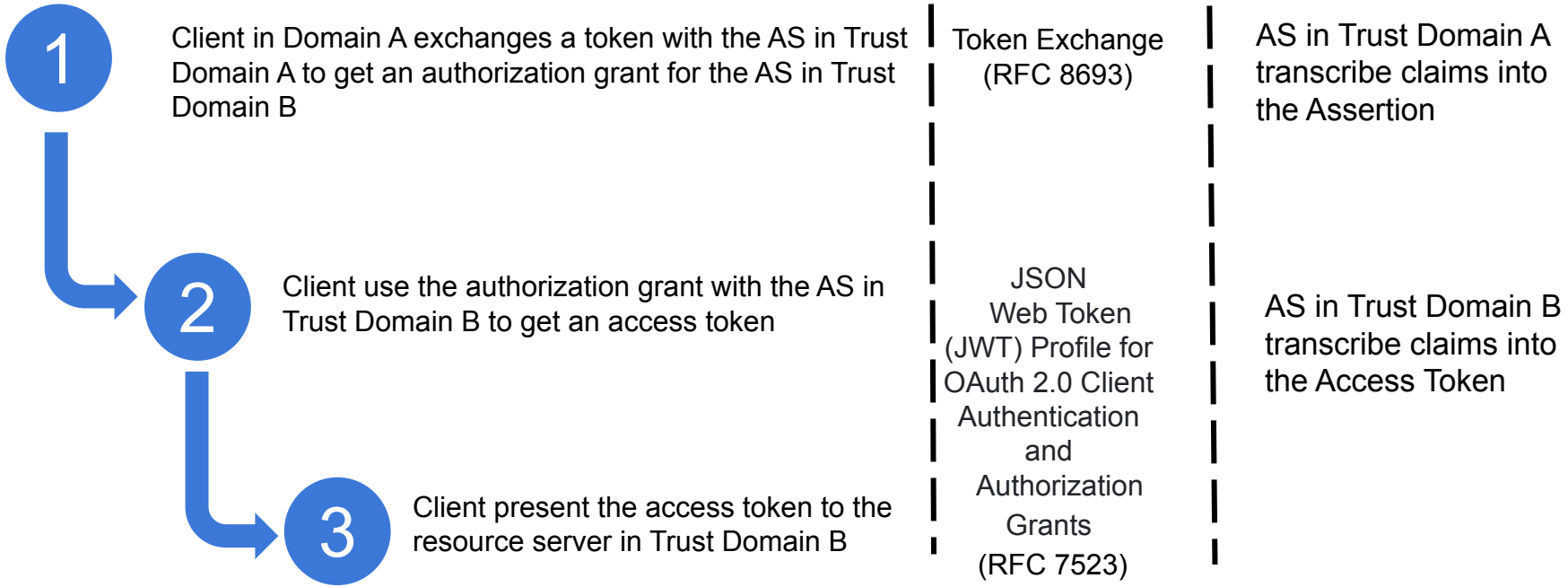


Image Courtesy Justin Richer (justin@bspk.io)

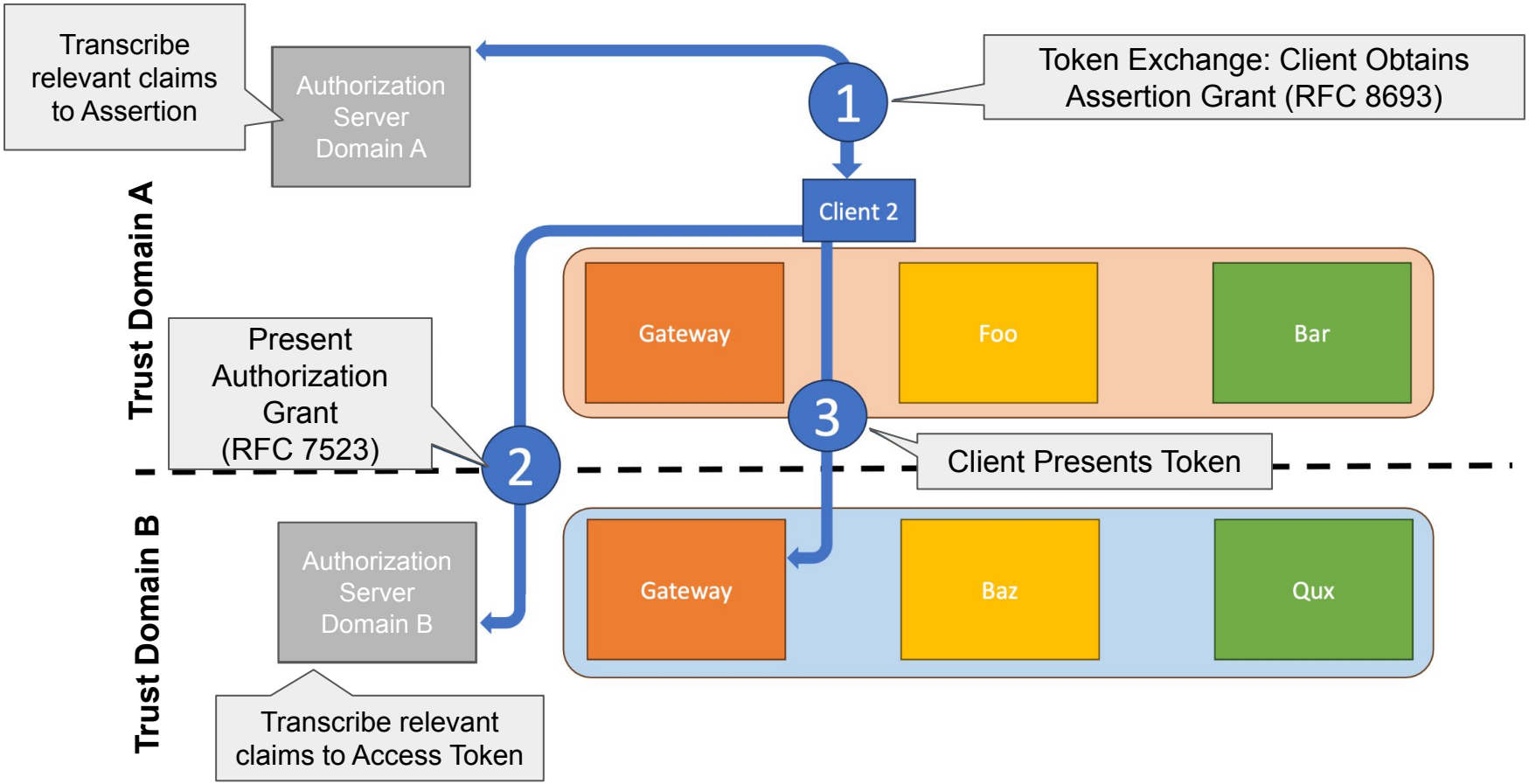
RFC 8693 and RFC 7523 gives us the protocol mechanics



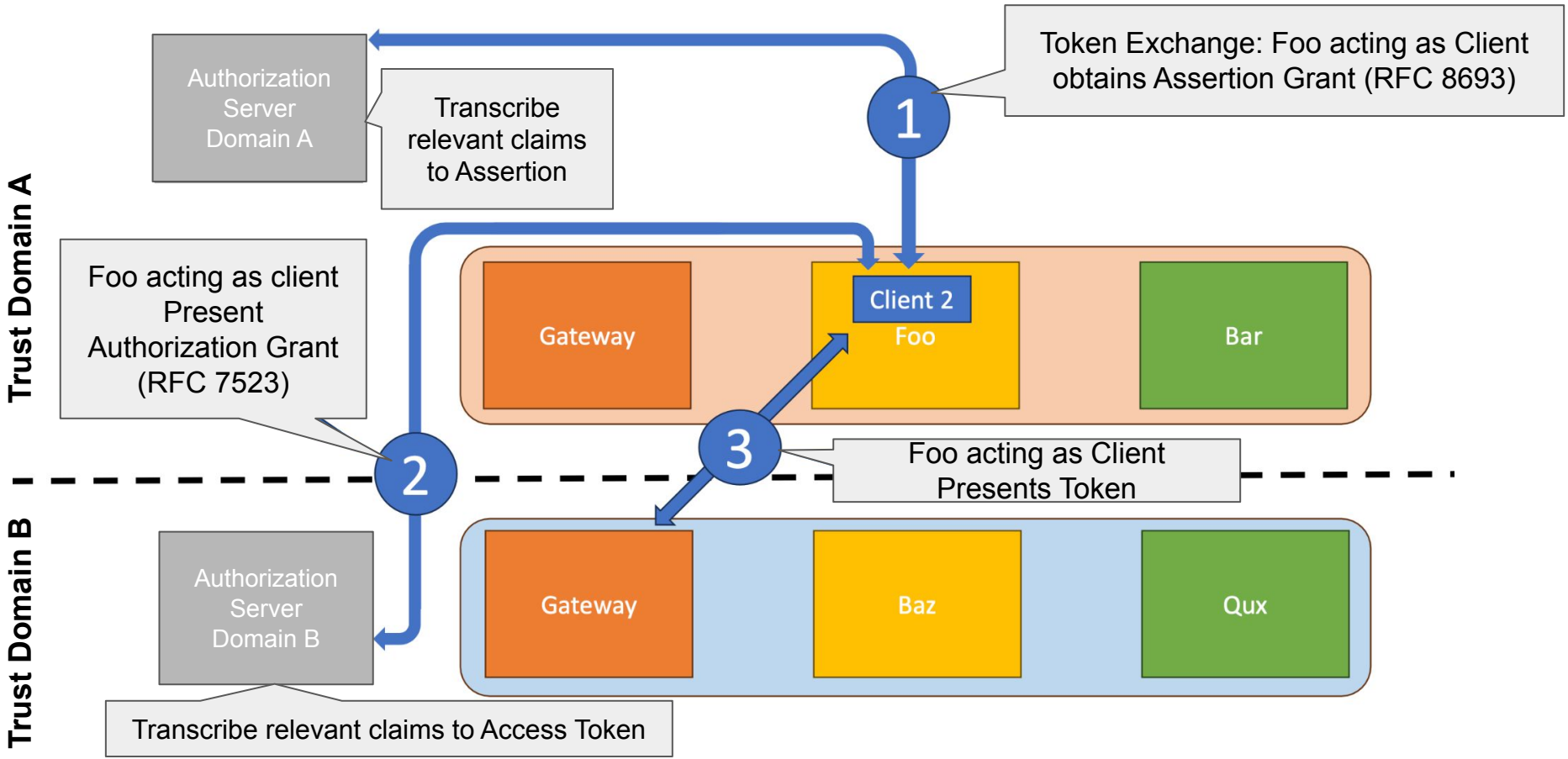
Authorization Servers can transcribe claims to preserve context



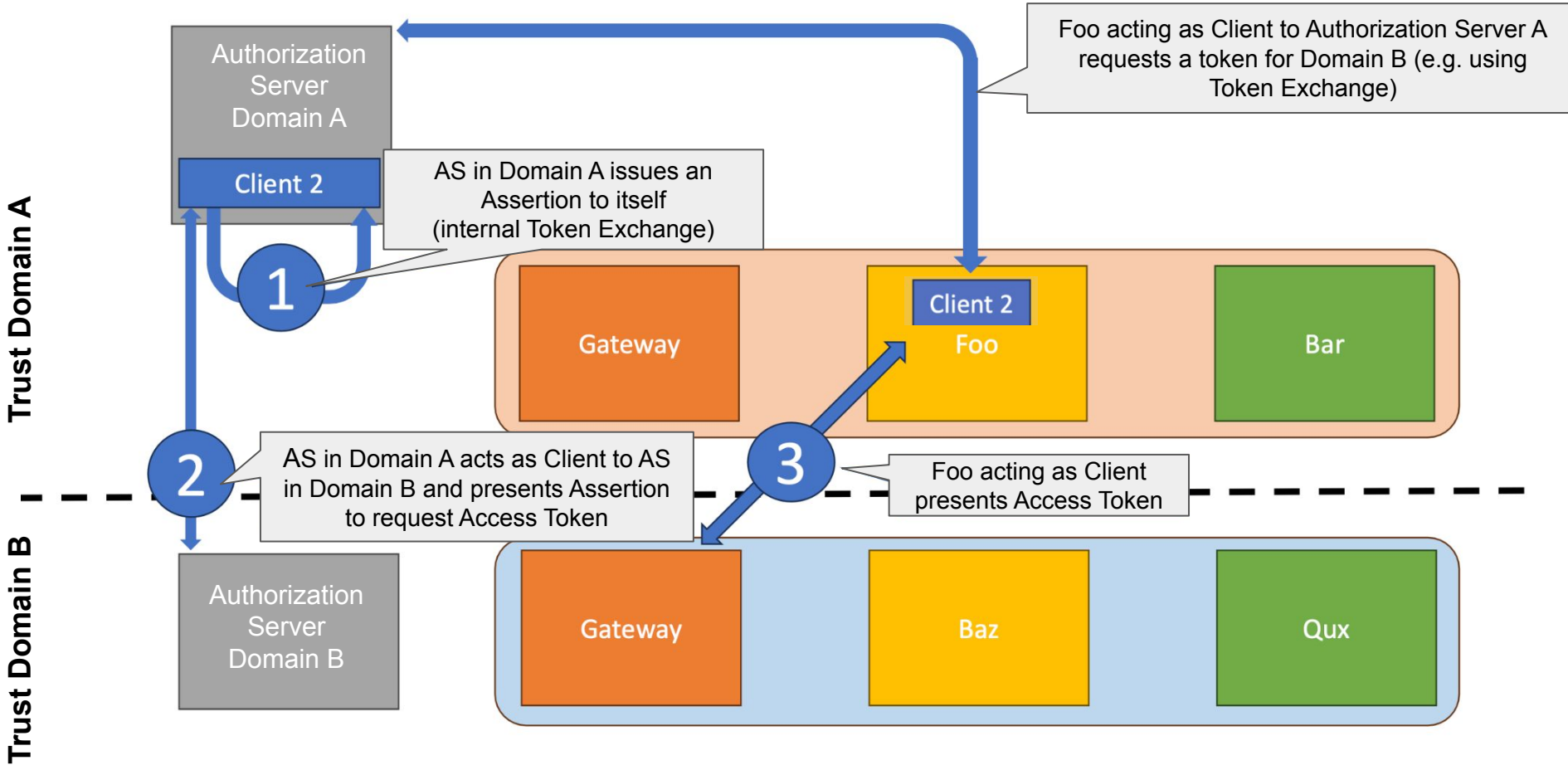
Cross Domain Identity Chaining



Use Case 1: Resource Server As Client

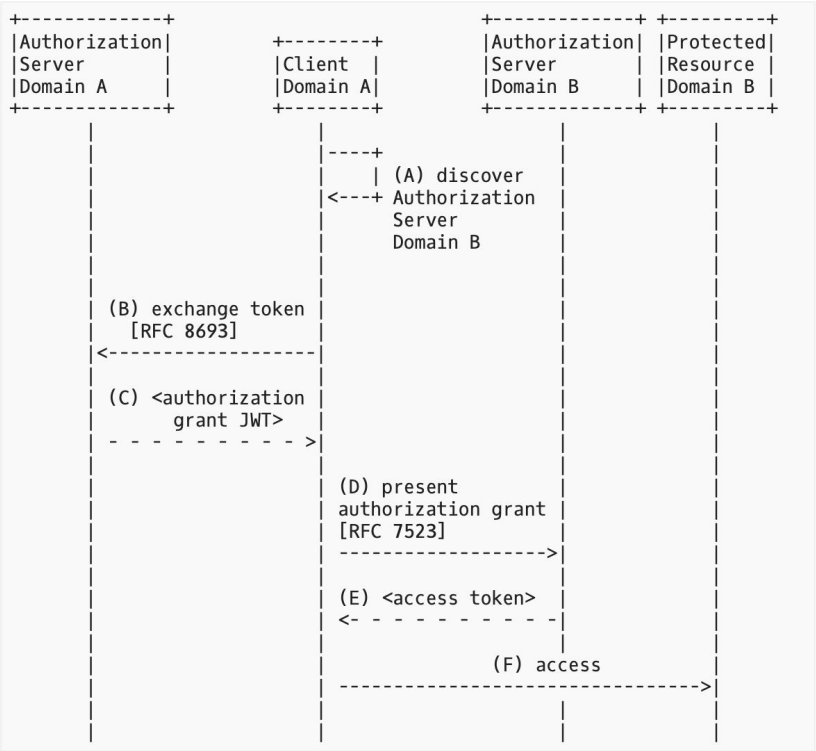


Use Case 2: Authorization Server As Client

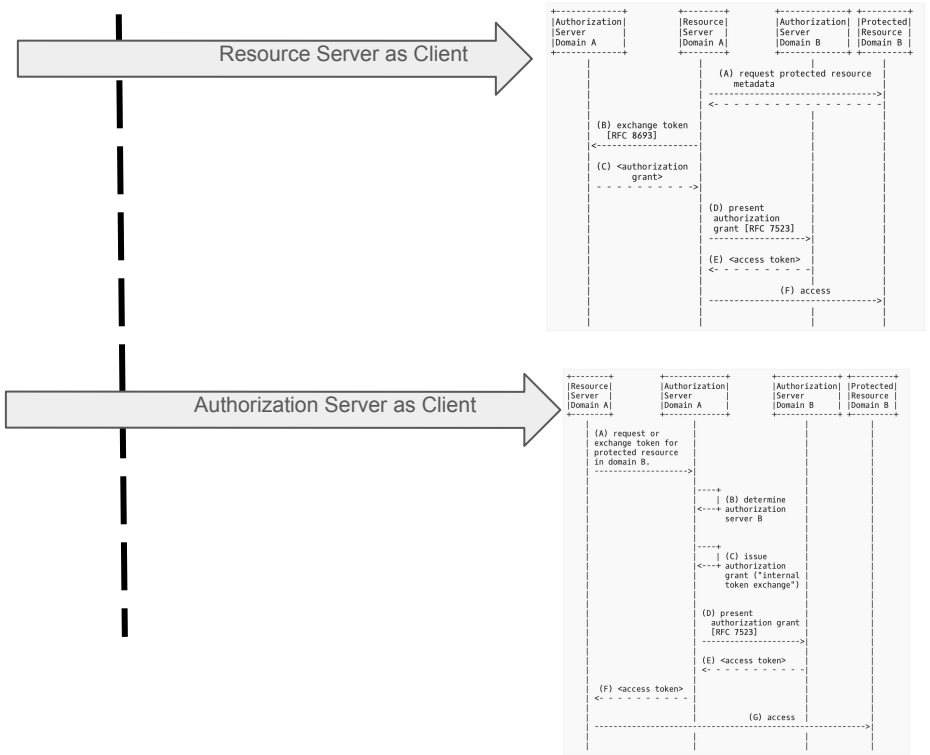


What's in the Draft?

Identity Chaining: Main Document (Normative)



Use Cases: Appendix (Non-Normative)



Open Question: Sender Constraining Tokens

Cross Domain Identity Chaining

1. Client requesting Access Token interact with AS in Domain B
2. **Use MTLS or DPoP to constrain tokens** ✓

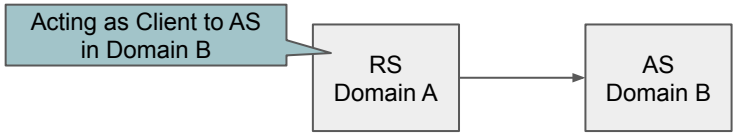
Open Question: Sender Constraining Tokens

Cross Domain Identity Chaining

1. Client requesting Access Token interact with AS in Domain B
2. **Use MTLS or DPoP to constrain tokens** ✓

Use Case 1: Resource Server as Client

1. Resource Server requesting Access Token while acting as client interacts with AS in Domain B
2. **Use MTLS or DPoP to constrain tokens** ✓



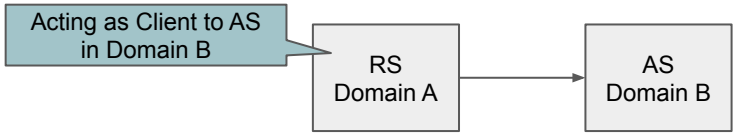
Open Question: Sender Constraining Tokens

Cross Domain Identity Chaining

1. Client requesting Access Token interact with AS in Domain B
2. **Use MTLS or DPoP to constrain tokens** ✓

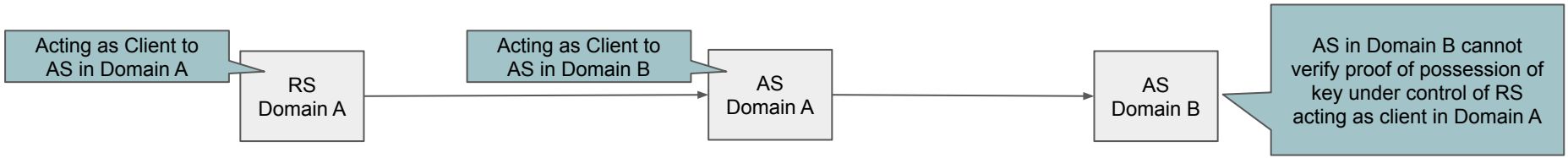
Use Case 1: Resource Server as Client

1. Resource Server requesting Access Token while acting as client interacts with AS in Domain B
2. **Use MTLS or DPoP to constrain tokens** ✓

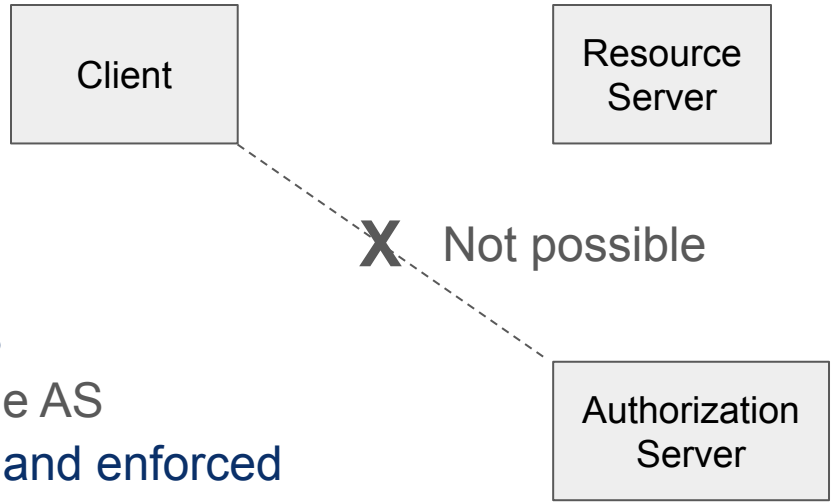


Use Case 2: Authorization Server as a Client

1. Resource Server requesting an Access Token does not interact directly with AS in Domain B.
2. Authorization Server in Domain A acts as client and requests an Access Token on-behalf-of the Resource Server in Domain A.
3. **Not supported by MTLS or DPoP -> A new "Delegated Key Binding Mechanism"?**



The Client cannot access the Authorization Server

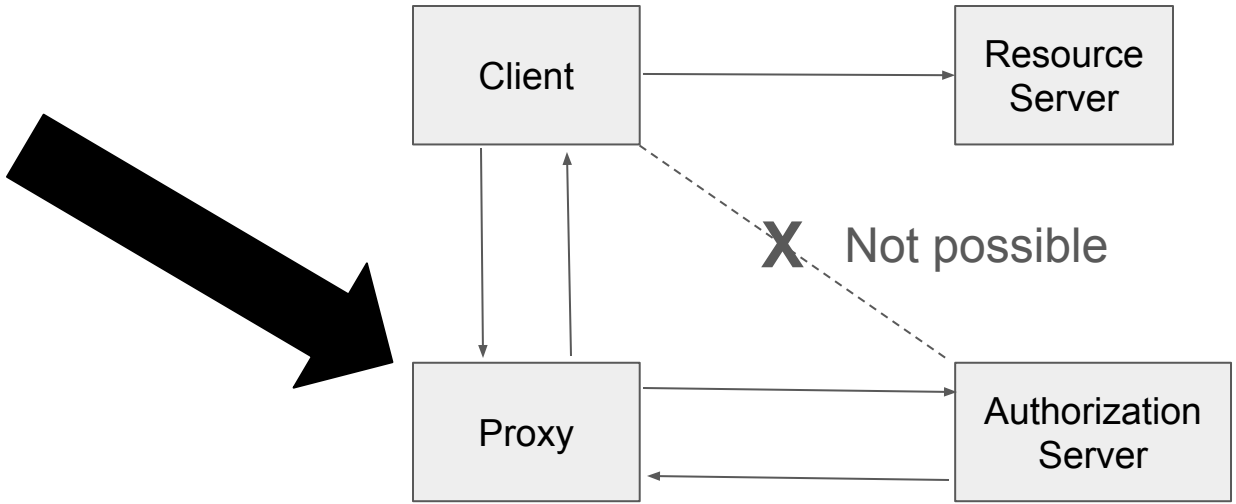


Why?

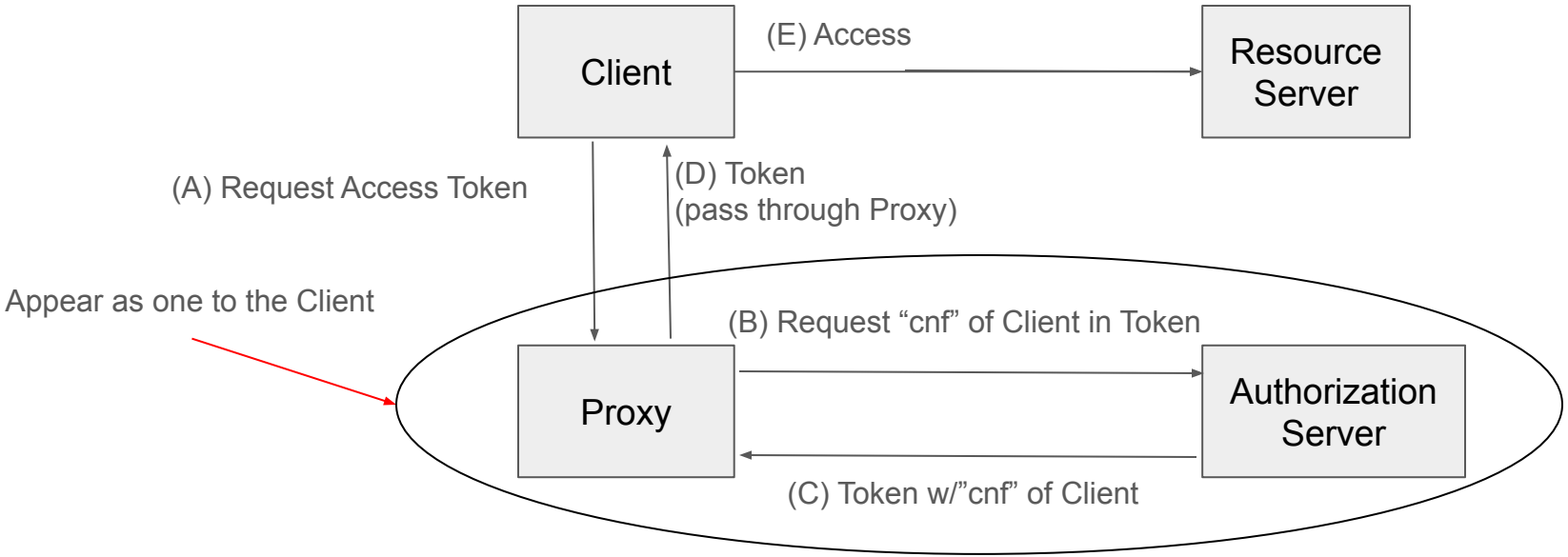
- The Client may not have knowledge of the AS
- The Client may not have network access to the AS
- Strict access control on resources is required and enforced by the AS
- The AS requires authentication, but the Client cannot

A proxy can access the Authorization Server

Proxy does have Access!

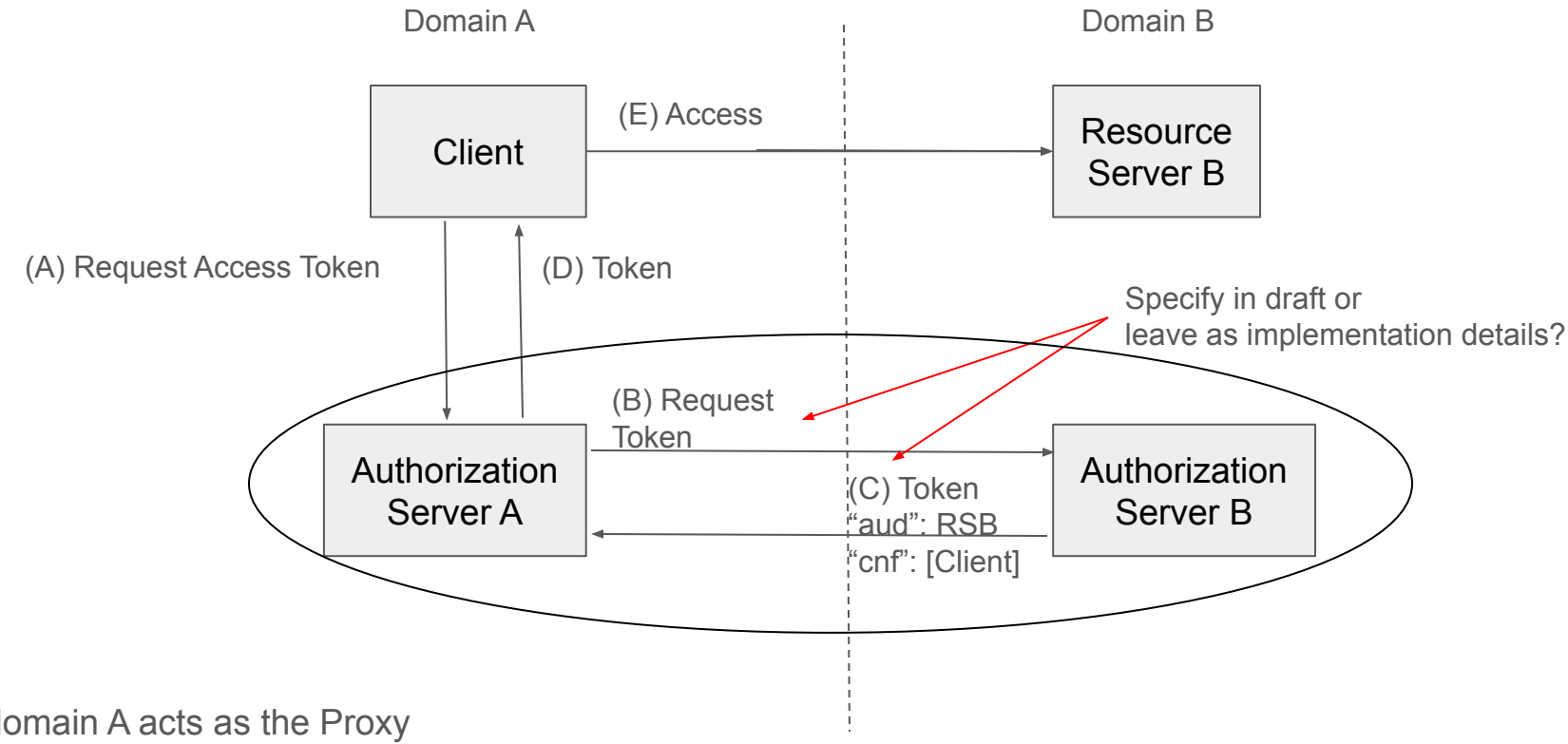


Delegated sender constraining of tokens (Key Binding)



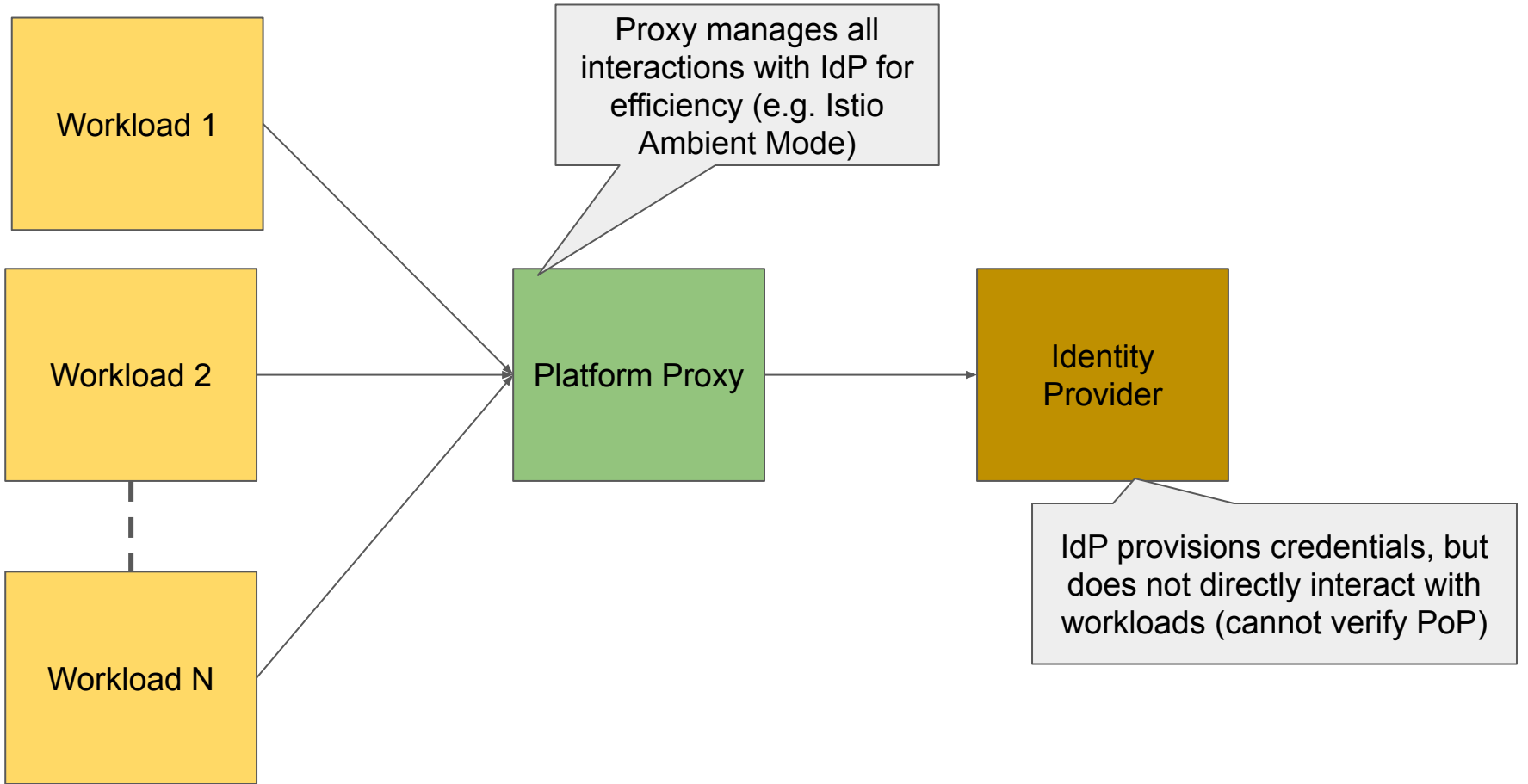
- Proxy, AS seem as one entity to the Client
- Proxy has trust relationship with AS
- Proxy verifies PoP from Client
- Proxy forwards Client’s “cnf” claim to AS
- AS includes Client’s “cnf” claim in access token

Use Case #2 (in Identity Chaining draft)



- AS in domain A acts as the Proxy
- Trust Relationship between AS in Domain A and AS in Domain B

Other Applications Emerging



Questions for Working Group

Sender Constraining tokens:

1. Should all use cases be supported (including the “AS as Client Proxy”)
2. Should sender constraining be described for the Identity Chaining flow and “RS as Client” use case (using MTLS and DPoP)?
3. Should sender constraining tokens be omitted from this draft and left for profiles or future drafts?

Delegated Key Binding

4. Should Delegated Key Binding be left as an implementation detail?
5. Should Delegated Key Binding be defined in this draft?
6. Should Delegated Key Binding be defined in another draft?