IETF 118 OAuth Working Group Meeting

Arndt Schwenkschuster (Microsoft) Pieter Kasselman (Microsoft) Kelley Burgin, (MITRE) Mike Jenkins (NSA-CSS) Brian Campbell (Ping)



### Agenda

- The challenge of Identity Chaining
- A (proposed) approach
- What's in the draft
- Next Steps



Trust Domain 1

**Trust Domain 2** 



Trust Domain 1

2

Domain

Trust



Trust Domain 1

**Trust Domain 2** 



## Proposal Concepts

### Getting an Authorization Grant for another Trust Domain

Client in Domain 1 exchanges a token with the AS in Trust Domain 1 to get an authorization grant for the AS in Trust Domain 2

Token Exchange ( RFC 8693)

Client use the authorization grant with the AS in TrustAssertionDomain 2 to get an access tokenFramework

(RFC7521)



Client presents the access token to the Resource server in Trust Domain 2

### Generic Cross-Domain Identity Chaining



### Resource Server as Client



Trust

### Authorization Server as Client



Trust Domain 2



## What's in the draft

https://datatracker.ietf.org/doc/draft-identity-chaining/

### Section 2.2 Generic Cross-Domain Identity Chaining

++		+	+ +
Authorization	++	Authoriza	tion   Protected
Server	Client	Server	Resource
Domain A	Domain A	Domain B	Domain B
++	++	+	+ +
	+		
	A)	) discover	
	<+ Au	thorization	
	Se	rver	
			1
(B) exchan	ge token	i i	
[RFC 869	3]		l l
<			1
(C) <autho< td=""><td>rization  </td><td></td><td></td></autho<>	rization		
gran	t>		
	>		
		sent l	
	authori	zation grant	
	[RFC 75	211	
		>	i i
i i		i i	i
i i	(E) <ac< td=""><td>cess token&gt;</td><td>ĺ</td></ac<>	cess token>	ĺ
1	<		1
l l			1
		(F) acces	s
			>

Figure 1: Identity Chaining Flow

### Appendix A.1 **Resource Server** acting as Client

+	+		+	+		+	+	+	
Authoriz	zation		Reso	urce		Author	ization	Prote	ected
Server			Serv	er		Server		Resou	irce
Domain A	A		Doma	in A		Domain	В	Domai	in B
+	+		+	+		+	+	· +	
									1
				(A) 1	request	protect	ted reso	ource	
				me	etadata	1			
								>	
				<					
	(C) ex	change tok	en						
	[RFC	8693]							
<	<						1		
	(D) <a< td=""><td>uthorizati</td><td>on  </td><td></td><td></td><td></td><td></td><td></td><td></td></a<>	uthorizati	on						
		grant>							
			>				1		1
				(E) pre	asont		1		1
				author	rizatio	n	1		1
				arant	IREC 7	75211	1		1
			- i			>	1		
i i			- i				i i		
i i			- i	(F) <ac< td=""><td>ccess t</td><td>oken&gt;</td><td>i</td><td></td><td></td></ac<>	ccess t	oken>	i		
i i			i	<			i		
i i			i				Ì		
i i	i					(G) ac	cess		
							>		
			- I				I		
									1

+--Au

+--

### Appendix A.2 Authorization Server acting as Client

++	+		+	+	+	++
Resource	Authori	ization		Author:	ization	Protected
Server	Server			Server	1	Resource
Domain A	Domain	Α		Domain	В	Domain B
++	+		+	+	+	++
						1
(A) request or						
exchange token	for	ĺ			ĺ	i i
protected reso	urce					i i
in domain B.						
	>					
		+				i i
			(B) determ	ine		
i i		<+ authorization				1
			server B			1
i i		i			İ	i i
						i i
		+				
			(C) issue			1
		<+	authorizat	ion		
			grant ("in	ternal		
i i			token exch	ange")	Ì	i i
i i		ĺ			Ì	i i
i i					İ	i i
i i		(D) p	resent			i i
i i		aut	horization	grant	İ	i i
i i		[ [RF	C 7521]	-	Ì	i i
i i				>	Ì	i i
		ĺ			Ì	i i
i i		(E) <	access tok	(en>	İ	i i
i i		<			Ì	i i
i i		ĺ			Ì	i i
(F) <access t<="" td=""><td>oken&gt;</td><td>i</td><td></td><td></td><td>İ</td><td>i i</td></access>	oken>	i			İ	i i
<		ĺ			I	i i
i					i	i i
			(G) a	ccess		i
						>
					1	i
						i
the second second second second second second second second second second second second second second second se						

## Token Exchange Profile

#### 2.4. Token Exchange

The client performs token exchange as defined in [RFC8693] with the authorization server for its own domain (e.g., Domain A) in order to obtain an authorization grant that can be used with the authorization server of a different domain (e.g., Domain B) as specified in section 1.3 of [RFC6749].

#### 2.4.1. Request

The parameters described in section 2.1 of [RFC8693] apply here with the following restrictions:

requested\_token\_type OPTIONAL according to [RFC8693]. In the context of this specification this parameter SHOULD NOT be used. See Authorization grant type (Section 2.4.3).



**Open Question:** Should this be restricted to JWT?

### Assertion Flow Profile

#### 2.5. Authorization Grant

The client presents the authorization grant it received from the authorization server in its own domain and presents it to the authorization server in the domain of the resources server it wants to access as defined in the "Assertion Framework for OAuth 2.0 Client Authentication and Authorization Grants" [RFC7521].

#### 2.5.1. Request

If the authorization grant is in the form of a JWT bearer token, the client SHOULD use the "JSON Web Token (JWT) Profile for OAuth 2.0 Client Authentication and Authorization Grants" as defined in [RFC7523]. Otherwise, the client SHOULD request an access token using the "Assertion Framework for OAuth 2.0 Client Authentication and Authorization Grants" as defined in [RFC7521] (Section 4.1). For the purpose of this specification the following descriptions apply:

#### grant\_type

REQUIRED. In context of this specification clients SHOULD use the type identifier returned by the token exchange (issued\_token\_type response). See authorization grant type (Section 2.4.3) for more details.

#### assertion

REQUIRED. Authorization grant returned by the token exchange (access\_token response).

#### scope

OPTIONAL.

The client MAY indicate the audience it is trying to access through the scope parameter or the resource parameter defined in [RFC8707].

### **Claims Transcription**

#### 2.6. Claims transcription

Authorization servers MAY transcribe claims when either producing authorization grants in the token exchange flow or access tokens in the assertion flow.

- \* \*Transcribing the subject identifier\*: Subject identifier can differ between the parties involved. For instance: A user is known at domain A by "johndoe@a.org" but in domain B by "doe.john@b.org". The mapping from one identifier to the other MAY either happen in the token exchange step and the updated identifer is reflected in returned authorization grant or in the assertion step where the updated identifier would be reflected in the access token. To support this both authorization servers MAY add, change or remove claims as described above.
- \* \*Selective disclosure\*: Authorization servers MAY remove or hide certain claims due to privacy requirements or reduced trust towards the targeting trust domain. To hide and enclose claims [I-D.ietf-oauth-selective-disclosure-jwt] MAY be used.
- \* \*Controlling scope\*: Clients MAY use the scope parameter to control transcribed claims (e.g. downscoping). Authorization Servers SHOULD verify that requested scopes are not higher priveleged than the scopes of presented subject\_token.
- \* \*Including authorization grant claims\*: The authorization server performing the assertion flow MAY leverage claims from the presented authorization grant and include them in the returned access token. The populated claims SHOULD be namespaced or validated to prevent the injection of invalid claims.

### Controlled by Authorization Servers

- 1. Subject identifier change
- 2. Selective disclosure
- 3. Controlling scope/down-scoping
- 4. Preserving claims

### **Open Question:**

Should we define how the claims are transcribed?

## Changes since IETF 117

- <u>Update docname to draft-schwenkschuster-oauth-identity-chaining-</u> <u>00</u>
- Editorial: Remove repetitive text
- <u>Replace cURL commands with "on-the-wire" examples</u>
- Add correct reference for RFC 7523
- <u>Clarify requirements for "aud" claim</u>
- Update Acknowledgements
- <u>Correct/Update Authorization Server Discovery</u>

## Next steps



Scope

- Consider limiting token formats to JWT
- How to transcribe claims



https://github.com/arndt-s/ietf-identity-chaining/issues

### Next Steps

• Interest in the WG to pursue this work?

# Questions?