Token and Identity Chaining Between Protected Resources Using OAuth Token Exchange

July 25, 2022
Kelley Burgin, kburgin@mitre.org
Background

- Published OAuth and OIDC profiles to enable “identity bridging” and federated authentication
  - Identity of user and client are bridged to access a resource
- Published Token and Identity Chaining Profiles (drafts)
  - Single and multi-ICAM environment use cases
  - Update removes some options, presents a new solution
Token and Identity Chaining in a Multi-ICAM Environment

Problem:
- In a multi-ICAM ecosystem, an OAuth client makes a request to a protected resource PR1 in its organization, but PR1 needs to access a second PR2 in a different organization to answer the client's request.

Solution:
- PR1, acting as an OAuth client, uses the IETF OAuth Token Exchange protocol to exchange received access token with its AS1 for a new access token that it can use to access PR2.

Goals for this week:
- Get feedback on Token Chaining Profiles
- Get feedback on newly defined claim (chained_id)
Relevant Token Chaining Profile Requirements

- The new access token received by PR1 from token exchange
  - Has PR1 as the "client_id"
  - Is sender-constrained to PR1’s PKI certificate using "cnf" claim
  - Contains an "act" claim with:
    - "sub" claim identifying PR1
    - "iss" claim identifying the AS generating the token
    - All "act" claims from previous tokens to show entire history of token holders back to the original client

  For verification by PR2

  PR2 may use for authorization decisions
Multiple ICAM Ecosystem – Option 1
AS1 Generates the New Token

Summary:
- PR1 performs token exchange with AS1
- AS1 generates the new token
- PR1 presents the token to PR2
- PR2 presents token to AS2
- AS2 validates the token with AS1 out-of-band using the attribute sharing infrastructure
- PR2 returns the requested data to PR1
Multiple ICAM Ecosystem – Option 2
AS2 Generates the New Token

Summary:
- PR1 performs token exchange with AS1
  - AS1 generates a JWT assertion that it uses to obtain the access token from AS2
  - AS2 generates the token and returns it to AS1, who returns it to PR1 to complete the token exchange request

Problem:
- We need PR1 “client_id” and “cnf” fields in the token for PR2 to verify
- So AS1 needs to pass these two bits of information to AS2 in its request to AS2 for the token
Multiple ICAM Ecosystem – Option 2
AS2 Generates the New Token

Solution:

- Define a new private use OAuth claim
  
  `chained_id {`
  
  "client_id": "PR1"
  
  "cnf": [Hash of PR1 PKI certificate]
}

- AS1 includes “chained_id” in its token request to AS2
- AS2 populates new token with “iss”, “exp”, “sub”, “aud” according to specs and/but
  
  "client_id" and “cnf” claims are populated with the values of PR1 obtained in the “chained_id” claim.
What’s Next

- Implementations
  - Ping Federate complete
    - Custom processing at AS1
  - Keycloak next
- Thoughts on new claim?
- Are these profiles generally useful?