GNAP WG

Evaluation and Recommendations on XYZ and XAuth proposals

Kathleen Moriarty
## Evaluation

<table>
<thead>
<tr>
<th>XAuth</th>
<th>XYZ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scope defined, abstract aimed at those with high-level of technical depth</td>
<td>Clear scope defined in abstract aligned to full OAuth scope</td>
</tr>
<tr>
<td>Intended as a next generation OAuth, but makes transition a lower impact, not fully backwards compatible</td>
<td>Next generation OAuth, revamping protocol. Not backwards compatible.</td>
</tr>
<tr>
<td>Interaction flows documented along with required fields</td>
<td>Interaction flows defined, focus more on security with cryptographic requirements and examples included</td>
</tr>
<tr>
<td>Relies heavily on OAuth2.0, using Bearer token and adding cryptographic (e.g. JOSE) functions after-the-fact</td>
<td>Builds security into the protocol as opposed to adding it in later (e.g. OAuth2.0 bearer token + JWT)</td>
</tr>
<tr>
<td>Defined terms and interactions comprehensive, supporting use cases of OAuth and OpenID Connect</td>
<td>Defined terms and interactions comprehensive, yet simplified</td>
</tr>
</tbody>
</table>
### Evaluation Continued

<table>
<thead>
<tr>
<th>XAuth Relationships</th>
<th>XYZ Relationships</th>
</tr>
</thead>
<tbody>
<tr>
<td>User, Client, Registered Client, Dynamic Client</td>
<td>Resource Client (abstracted term to cover both user &amp; client )</td>
</tr>
<tr>
<td>Grant Server</td>
<td>Authorization Server</td>
</tr>
<tr>
<td>Resource Server</td>
<td>Resource Server</td>
</tr>
<tr>
<td>Resource Owner</td>
<td>Resource Owner</td>
</tr>
</tbody>
</table>

- XYZ's simplified set of roles abstracted allow for the interactions described in XAuth.
- XAuth defines 4 sequences using the defined relationships with diagrams,
- XYZ defines one set of interactions that may include various sequences.
- The terms sequences and API in XAuth maps to the protocol functionality in XYZ that begin at section 2.

- Maintain the simplified set of roles,
- Improve/add the interaction models adding diagrams in XYZ to establish architectural patterns.
Evaluation continued

- XAuth API
  - Read, write, and other object interactions defined including sets of fields required

- XYZ
  - Protocol interactions listed in terms of functionality
    - Request access, request resources, etc.
    - Identify client, user, etc.
    - Content requirements for fields defined in this context
    - Extensive set of interactions defined, building on field definitions

- The two proposals diverge heavily in the API/interactions definition style
  - XYZ has defined more specific interactions, allowing for extensibility
  - XYZ’s interactions are further developed
  - XAuth has a solid base to build upon using a different style to establish the API
Evaluation Continued

- Both have thrown a dart to deal with authentication later
  - Recommend use of a registry to track with authentication protocol preferences (e.g. security level, pointer to RFC and security considerations)
  - Authentication is required, correctly handled by another protocol(s)
- Cryptographic functions
  - XAuth uses JWTs as is done in OAuth for security, where as
  - XYZ embeds the cryptographic functions within the defined exchanges directly, sometimes using a JWT.
- Both allow for extensibility, which should be maintained
Recommendations

- Select one option to develop within the WG
  - WG name does not have to match protocol name
  - Adding a new protocol name to the mix is confusing for this already flooded space.

- OAuth2.0 has known limitations and security protocol proofing demonstrated the deficiencies

- Develop next generation OAuth with security researcher involvement
  - Iterate on security protocol proofing and evaluations
  - Backwards compatibility should not take precedence over security
  - Industry is trending towards building security in and zero-trust models
  - Simplify to improve ability to do threat modeling and security protocol proofing

- Compatibility
  - Ensure supported authentication mechanisms and strength of those solutions is clearly presented (use a registry)
  - Federation, scoping, and other features that build on an authorization protocol may evolve too
  - Maintain extensibility to aid adoption

- Editors
  - WG Chairs select the editors of adopted drafts, consider adding an editor
  - Resulting document is consensus driven.

- Issue Tracking
  - Once adoption has occurred, issue tracking to document consensus moving iteratively through issues to aid progress
Recommendations

- Adopt XYZ as a WG draft
  - Baseline document to be further developed by WG
  - Provides a well developed starting point with room for WG interaction to guide further development
  - Merge in XAuth diagrams, modifying to abstracted roles of XYZ, adding sequences as needed
  - Articulate the request/response requirements more clearly as is done in XAuth for the API
    - Xauth lists these similar to how “messages” are defined in other protocols