OAuth Security

Phil Hunt, Hannes Tschofenig
Status

• The charter has a security related item.
• In the meanwhile we had produced a more comprehensive threats and security requirements document:
Security and Privacy Threats

• List of threats is based on NIST Special Publication 800-63.
  – Token manufacture/modification
  – Token disclosure
  – Token redirect
  – Token reuse

• Details in Section 3 of
  
Threat Mitigation

• An important part of the threat mitigation is the protection of the token. This work was done in JOSE, in a separate working group, but originated in the OAuth WG.

• There are different directions regarding the mitigation of threats. Three broad classes exist:
  1. Confidentiality Protection
  2. Sender Constraint
  3. Key Confirmation

• RFC 6749 offers a solution to these threats using approach (1).

• Now, we tackle approach (3).
Security Requirements

• There are two components that need to be considered:
  – Client<->Authorization Server: Requesting and obtaining keying material and meta-data.
  – Client<->Resource Server: Confirming knowledge of the key
Privacy & Security Requirements, cont.

- **RFC 4962** provides guidance for three party authentication and key exchange protocols. Provides a good starting point.

- Requirements:
  - Cryptographic Algorithm Independent
  - Strong, fresh session keys
  - Limit Key Scope
  - Replay Detection Mechanism
  - Authenticate All Parties
  - Authorization
  - Keying Material Confidentiality and Integrity
Privacy & Security Requirements, cont.

• Requirements (cont.):
  – Confirm Cryptographic Algorithm Selection
  – Uniquely Named Keys
  – Prevent the Domino Effect
  – Bind Key to its Context
  – Authorization Restriction
  – Client Identity Confidentiality
  – Resource Owner Identity Confidentiality
  – Collusion
  – AS-to-RS Relationship Anonymity

• Details are provided in Section 5 of draft-tschofenig-oauth-security-00.txt
Suggested Design Approach

• Maximum re-use of available OAuth & JSON WG specifications.

• Develop two alternative solutions based on symmetric as well as asymmetric cryptography.
  – Hard to decide without being able to judge the details.

• Avoid options as much as possible.

• Tighten the usage of OAuth 2.0 features.
  – Mandatory client authentication.
  – Mandatory state attribute.

• Produce running code in parallel to the specification development.

• Chairs are considering conference calls for faster progress.
Strawman

• Client asks AS for a Token. Additional information, such as
  – Intended recipient
  – Scope
  – Algorithm indication
• Authorization Server returns two elements:
  – For Client consumption: Keying material, lifetime, key id, granted scope, and other authorization information relevant for the client.
  – For RS consumption: Access Token (with keying material included).
• Request and response encoded in JSON and is protected.
Strawman, cont.

• Client needs to demonstrate possession of a secret to the RS.
  – Creates JSON object including key-id, algorithm information, replay protection information.
  – Access Token also provided

• RS processes request and may derive keying material for subsequent Client<->RS interaction.

• TLS channel binding support provided, and can be added.