Auditable Authorization

Igor Zboran
izboran@gmail.com

***Abstract* – Bearer tokens are vulnerable at rest and in transit when an attacker is able to intercept a token to illegally access private information. In order to mitigate some of the risks associated with bearer tokens, an Authorization Audit Trail (AAT) may be used alongside the bearer tokens during the authorization process. AAT consists of cryptographically chained blocks of data bearing a chronological tamper-resistant records of all their possessors and the changes that have been made by them. A nested, chained MAC construction (e.g., HMAC) is used to ensure the authenticity and integrity of AAT. All sensitive information are encrypted to preserve confidentiality. The access control relies on a real-time auditability of AAT by the authorization server. The AAT concept is compatible with existing OAuth2 and UMA protocols.**

I. Introduction

Bearer tokens are easy to use and easy to integrate with any client, server, and mobile device. Once they have been minted on the authorization server, they don't require additional processing on the client, and token validation on the resource server is trivial. While existing technology standards are acceptable for many use cases, they have been harder adopt in the healthcare industry and financial sector, where higher degree of authenticity and integrity of authorization process is required. An auditable authorization concept allows Identity and Access Management (IAM) systems to enhance token-based authorization mechanism while ensuring compliance with financial and legal regulations.

II. Concept

The auditable authorization concept is based on a verifiable tamper-resistant audit trail created by authenticated participants (authorization server, client, resource server) during the respective stages of the authorization process. The audit trail carries information in the form of a sequence of records organized into blocks. Each block comes from an individual participant – a block possessor. Records and blocks are chained using the MAC value of the previous record.

A. Record Chaining

To create a chain of records we use the HMAC chaining construct $MAC = \mathrm{HMAC}(K, \mathrm{HMAC}(MAC, m))$, broken down into individual MACs,

$MAC = \mathrm{HMAC}(MAC, m)$
$MAC = \mathrm{HMAC}(K, MAC)$

which forms the basis of the record chaining mechanism.

To simplify notation, we use the Double HMAC construct – a nested HMAC function, denoted by DHMAC, that takes 3 inputs (K, MAC, m) and outputs a message authentication code

$$MAC = \mathrm{DHMAC}(K, MAC, m) = \mathrm{HMAC}(K, \mathrm{HMAC}(MAC, m))$$

where K is the secret key, MAC is the input message authentication code, and m is the message to be authenticated.

B. Block Chaining

The block possessor must be registered at the authorization server (public clients can use dynamic registration to become confidential clients).

Each block contains four mandatory records:

- The random NONCE to prevent replay attack.
- The timestamp of when the block was created.
- The URI that identifies who created the block.
- The MAC value of the last record from the previous block.

Additional groups of optional records can be added at any time until the block is sent to the next possessor.

C. Verification

Blocks are verified via the introspection endpoint of the authorization server.

III. Conclusion

By utilizing simple cryptographic techniques, the AAT mechanism may be used alongside the bearer tokens during the authorization process. This concept of auditable authorization mitigates the risk associated with bearer tokens to illegally access private information. In a broader scope of the auditable authorization, there is the possibility of using the recorded data for forensic analysis and verification of legal compliances.

IV. Future Work

As a part of future work, we plan to explore the possibility of replacing the bearer tokens with AAT and storing JWT claims in the form of a set of records directly into chained blocks.

Acknowledgment

Credits go to WG - User-Managed Access, Google Research Publications and Audit in OAuth 2.0.