OAuth Working Group
Internet-Draft
Intended status: Standards Track
Expires: December 31, 2015

N. Sakimura
Nomura Research Institute
K. Li
Alibaba Group

June 29, 2015

Sender Constrained JWT for OAuth 2.0
draft-sakimura-oauth-rjwtprof-04

Abstract

This discussion document describes a method to indicate a sender constraint within JWT. It could potentially be incorporated into POPS spec [POPS].

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on December 31, 2015.

Copyright Notice

Table of Contents

1. Introduction

OAuth 2.0 Proof-of-Possession (PoP) Security Architecture [POPA] identifies Sender Constraint and Key Confirmation as possible threat mitigation methods against the use of token by an unauthorized presenter. While Proof-Of-Possession Semantics for JSON Web Tokens (JWTs) [POPS] touches briefly on the Sender Constraint, it is only one paragraph within a introductory text and does not discuss it in detail. Instead, it devotes much of the discussion to the Key Confirmation method. It also is making the usage of such token against the resource server out of scope.

This discussion draft describes a way to express the Sender Constraint in the JWT, as well as one possible way of using it to access a protected resource.

1.1. Notational Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

Unless otherwise noted, all the protocol parameter names and values are case sensitive.

2. Terminology

For the purpose of this document, the terms defined in RFC6749 [RFC6749] is used. In addition, following term is defined.

Authorized Presenter Party that the token is intended to be used by.

3. Justification

There are scenarios that the bearer token may be stolen, modified, reused or replayed. To prevent these threats, resource servers need to obtain additional assurance that the client is indeed authorized to present an access token. The detailed use cases can be found in OAuth 2.0 Proof-of-Possession (PoP) Security Architecture [POPA] specificaiton.

As described in OAuth 2.0 Proof-of-Possession (PoP) Security Architecture [POPA] specificaiton, there are several ways to prevent these bearer token threats: Confidentiality Protection, Sender Constraint and Key Confirmation. Key Confirmation mechanism is desribed in OAuth 2.0 Proof-Of-Possession Semantics for JSON Web Tokens (JWTs) [POPS] specification in detail, but Sender Constraint mechanism is not explained in detail.

In fact, Key Confirmation mechanism increased a lot of complexity, and a complete key distribution protocol has to be defined. Sender Confirmation mechanism can be relatively easier to implement in some cases, for example, when the client identity informaiton is easy to be accessed through APIs, when the client authentication is easy to achieve. So, Sender Confirmation mechanism should also be specified, and it can work as an alternative mechanism to mitigate the bearer token threats.

4. Sender Constraint Representation

Sender Constraint is expressed by including the following member at the top level of JWT payload.

azp The Client ID of the Authorized Presenter.

Following is an example of such JWT payload.

```
{
 "iss": "https://server.example.com",
 "sub": "joe@example.com",
 "azp": "clientID-1342050",
 "aud": "https://client.example.org",
 "exp": "1361398824",
 "nbf": "1360189224",
}
```

Figure 1 Example of Sender Constrained JWT.

5. Client Authentication

The resource server that supports this specification MUST authenticate the Client. In this document a possible method is proposed as follows:

1. The authorized presenter issues a HEAD or GET request to the resource server.

```
GET /resource/1234 HTTP/1.0
Host: server.example.com
```

2. The resource server returns a HTTP 401 response with "WWW-Authenticate" header with "Named" scheme, whcih includes nonce.

```
HTTP/1.0 401 Unauthorized
Server: HTTPd/0.9
Date: Wed, 14 March 2015 09:26:53 GMT
WWW-Authenticate: Named nonce="dcd98b7102dd2f0e8b11d0f600bfb0c093"
```

3. The client creates JWS compact seriarization over the nonce.

4. The client sends the request to the resource server, this time with Authorization: header with Named scheme and access token and the JWS.

```
GET /resource/1234 HTTP/1.0
Host: server.example.com
Authorization: Named at="access.token.jwt", s="jws.of.nonce"
```

5. The resource server finds the client key corresponding to the value of "azp" in the access token. It may have been obtained through client registration at the Issuer.

6. The resource server creates the JWS of the nonce and compares it with the value of "s" of the Authorization header. If it fails, the process stops here and the resource access MUST be denied.

7. The resource server MUST verify the access token. If it is valid, the resource SHOULD be returned as HTTP resonse.

6. IANA Considerations

6.1. Named Authentication Scheme

A new scheme has been registered in the HTTP Authentication Scheme Registry as follows:

Authentication Scheme Name: Named

Reference: Section xx of this specification

Notes (optional): Tha Named Authentication scheme is inteded to be used only with OAuth Resource Access, and thus does not support proxy authentication.

6.2. JSON Web Token Claim Registration

This specification registers the Destination Claim defined herein in the IANA JSON Web Token Claims registry defined in [I-D.ietf-oauth-json-web-token].

6.2.1. Registry Request Contents

- Claim Name: "azp"
- Claim Description: The Client ID of the Authorized Presenter
- Change Controller: IESG
- Specification Document(s): Section 3 of this document

7. Security Considerations

Needless to say, the client's secret key must be kept securely.

8. Acknowledgements

TBD

9. References

9.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC6749] Hardt, D., "The OAuth 2.0 Authorization Framework", RFC 6749, October 2012.

9.2. Informative References

[POPA] Hunt, P., Ed., "OAuth 2.0 Proof-of-Possession (PoP) Security Architecture", March 2015.

[POPS] Jones, M., "Proof-Of-Possession Semantics for JSON Web Tokens (JWTs)", March 2015.

Appendix A. Document History

-04 Added justification section

-03 Removed most of the duplication with [POPS]

-02 Included key confirmation method etc. The first version on the tools.ietf.org. (Prevous versions were sent just as email attachements.)

Authors' Addresses

Nat Sakimura
Nomura Research Institute

Email: sakimura@gmail.com

Kepeng Li
Alibaba Group

Email: kepeng.lkp@alibaba-inc.com