JESUÏTES educació

DAWM07UF4

SERVEIS WEB

UF4.7.2 JWT

CFGS Desenvolupament d'Aplicacions Web
M07. Desenvolupament web en entorn servidor
Fundació Jesuïtes Educació - Escola del Clot
Sergi Grau sergi.grau@fje.edu

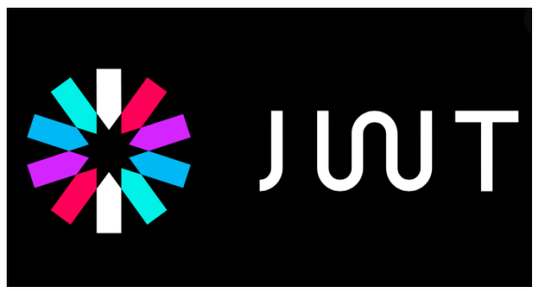
OBJECTIUS

- Conèixer que són els JSON web tokens i per a que s'utilitzen.

JSON WEB TOKEN

- JSON Web Token (JWT) és un estàndard obert (RFC 7519) que defineix una forma compacta i autònoma per transmetre informació de manera segura entre parts com a objecte JSON.

- Aquesta informació es pot verificar i confiar perquè està signada digitalment. Els JWT es poden signar mitjançant un secret (amb l'algorisme HMAC) o un parell de claus públiques / privades mitjançant RSA o ECDSA.



JSON WEB TOKEN

- El testimoni està firmat per la clau del servidor, així que el client i el servidor son tots dos capaços de verificar que el testimoni és legítim.

- Los JSON Web Tokens estan dissenyats per a ser compactes, poder ser enviats a les URLs -URL-safe- i ser utilitzats en escenaris de Single Sign-On (SSO).

- Els privilegis dels JSON Web Tokens poden ser utilitzats per propagar la identitat d'usuaris com a part del procés d'autenticació entre un proveïdor d'identitat i un proveïdor de servei, o qualsevol altre tipus de privilegis requerits per processos empresarials

ESTRUCTURA

- Els JSON web Tokens generalment estan formats per tres parts: una capçalera o **header** , un contingut o **payload** , i una signatura o **signature** . La **capçalera** identifica quin algoritme va ser usat per generar la signatura i normalment es veu de la següent manera

- header = '{ "alg": "HS256", "typ": "JWT"}'

- HS256 indica que aquest símbol està signat utilitzant HMAC-SHA256.

- El **contingut** conté la informació dels privilegis o claims de el testimoni:

- payload = '{ "loggedInAs": "admin", "iat": 1422779638}'

ESTRUCTURA

- L'estàndard suggereix incloure una marca temporal o **timestamp** en anglès, anomenat iat per indicar el moment en què el testimoni va ser creat.

- La **firma** està calculada codificant **l'encapçalament i el contingut en base64url, concatenándose ambdues parts amb un punt com a separador**:

- key = 'secretkey'

- unsignedToken = encodeBase64Url (header) + '.' + EncodeBase64Url (payload)

- signature = HMAC-SHA256 (key, unsignedToken)

ESTRUCTURA

- En el **token**, **les tres parts -encapçalat, contingut i signat estan concatenades utilitzant punts** de la següent manera:

- token = encodeBase64Url (header) + '.' + EncodeBase64Url (payload) + '.' + EncodeBase64Url (signature)

- El token és: eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJsb2dnZWRJbkFzIjoiYWRtaW4iLCJpYXQiOjE0MjI3Nzk2Mzh9.gzSraSYS8EXBxLN_oWnFSRgCzcmJmMjLiuyu5CSpyHI

- pot ser fàcilment transmès en entorns HTML i HTTP , sent similar a estàndards basats en XML com SAML

- En autorització s'aconsegueix quan l'usuari ingressa les seves credencials amb èxit, llavors **es genera un JSON web Token que és retornat a el client, qui ha de guardar localment**, **en comptes del model tradicional de crear una sessió en el servidor i de retornar una cookie** .

- Sempre que l'usuari vol accedir a una ruta protegida o recurs, **el client ha d'enviar el JWT**, generalment a la **capçalera de Authorization** utilitzant l'esquema Bearer. El contingut d'la capçalera HTTP es veu de la següent manera:

- Authorization: Bearer eyJhbGci...<snip>...yu5CSpyHI

- Aquest és un mecanisme **d'autenticació sense estat** - stateless - ja que la sessió de l'usuari mai es guarda en el proveïdor d'identitat o en el proveïdor de el servei. Els recursos protegits sempre comprovessin si hi ha un JWT vàlid en cada comanda d'accés. **Si el token és present i és vàlid, el proveïdor de el servei atorga accessos als recursos** protegits. Com **els JWTs contenen tota la informació necessària en si mateixos**, es redueix la necessitat de consultar la base de dades o altres fonts d'informació múltiples vegades.

codi	nom	Descripció
`iss`	issuer	Identifica el proveïdor d'identitat que va emetre el JWT
`sub`	Subject	Identifica l'objecte o usuari en nom de el qual va ser emès el JWT
`aud`	Audience	Identifica l'audiència o receptors per la qual cosa el JWT va ser emès, normalment el / s servidor / s de recursos (per exemple l'API protegida). Cada servei que rep un JWT per a la seva validació ha de controlar l'audiència a la qual el JWT està destinat. Si el proveïdor de el servei no es troba present en el camp `aud`, llavors el JWT ha de ser rebutjat
`exp`	Expiration time	Identifica la marca temporal després de la qual el JWT **no ha** de ser acceptat.
`nbf`	not before	Identifica la marca temporal en què el JWT comença a ser vàlid. EL JWT **no té** de ser acceptat si el testimoni és utilitzant abans d'aquest temps.
`iat`	Issued at	Identifica la marca temporal en què el JWT va ser emès.
`jti`	JWT ID	Identificador únic de el testimoni fins i tot entre diferent proveïdors de servei.

Els camps següents poden ser utilitzats a la capçalera:

codi	nom	Descripció
`typ`	token type	Si està present, es recomana utilitzar el valor `JWT`.
`cty`	content type	En casos normals, no és recomanat. En casos de signatura o xifratge niat, **ha de** està present i el valor ha de ser `JWT`.
`alg`	Message authentication code algorithm	El proveïdor d'identitat pot triar lliurement l'algoritme per a verificar la signatura d'el testimoni, encara que alguns dels algoritmes suportats són insegurs.

BIBLIOGRAFIA

+ https://jwt.io/

+ https://en.wikipedia.org/wiki/JSON_Web_Token