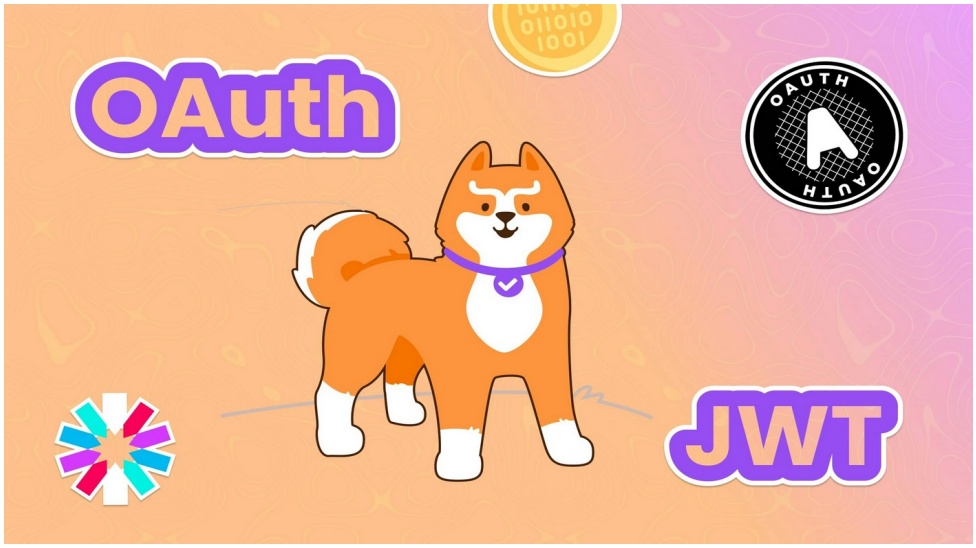
OAuth vs. JWT:

Quau ei la diferéncia entau desvolopament d'aplicacions

Exploratz JWT e OAuth deus ròtles distints dens la securitat de las aplicacions web, la faiçon dont tribalhan amassa e la lor importància dens lo desvolopament web modèrne.

Autor : Gabriel L. Manoir, Jan 08 2024

Revirada en occitan : Joan-Lucì Labòrda



Introduccion

Aquestas annadas, duas tecnologias de prumèr plan que son vadudas largament utilizadas dens la securitat de las aplicacions web: **OAuth** e **JSON Web Tokens (JWT**). Si los dus que jògan un ròtle essenciau dens los procediments d'autentificacion e d'autorizacion, que sèrven deus objectius distints e que foncionan segon principis diferents . Aqueth article que's penja sus las especificitats d'OAuth e de JWT, en comparant las lors foncionalitats, los lors cas d'utilizacion e la faiçon dont se completan dens la securizacion de las aplicacions web.

JWT: Lo geton autonòme

Lo JWT qu'ei un format compacte e securizat d'URL principaument utilizat entà transméter informacions enter las partidas en tota securitat. Qu'ei un objècte JSON qui pòt encoder ua varietat de revindicacions, talas com l'identitat e los atributs de l'utilizator. La beutat de las JWT que resideish dens la lor natura autonòma, qu'ei a díser que lo geton eth medish que contien totas las informacions necessàrias, qui son verificadas per signaturas numericas.

Entà compréner los JWT, qu'exploram la lor estructura, qui compren tres partidas: **l'entèsta, la carga utila e la signatura**. L'entèsta que declara generaument lo tipe de geton (JWT) e l'algoritme de signatura (per exemple, HMAC SHA256). La carga utila que contien las reclamacions, qui

pòden estar enregistradas, publicas o privadas. Fin finau, la signatura qu'ei calculada en codant l'entèsta e la carga utila dab l'ajuda de Base64URL, puish en la signant dab ua clau secreta.

Per exemple, que consideram un geton JWT utilizat dens un scenari d'autentificacion de l'utilizator:

```
// Header
{
  "alg": "HS256",
  "typ": "JWT"
}
// Payload
{
  "sub": "1234567890",
  "name": "John Doe",
  "admin": true
}
// Signature
HMACSHA256(
  *base64UrlEncode(header) + "." +
  base64UrlEncode(payload),
  secret)*
```

Aqueth exemple qu'illustra un geton JWT on la carga utila pòrta l'identitat, lo nom d'utilizator e los drets administratius deu subjècte. Lo servidor que pòt validar aqueth geton en utilizant l'algoritme e la clau secreta especificats.

OAuth 2.0: Lo quadre d'autorizacion delegat

OAuth, en particular OAuth 2.0, n'ei pas un format simbolic mes un quadre d'autorizacion. Que defineish ua seria de flux, o «tipes de subvencions», qui permeten a ua aplicacion clienta d'accedir a ressorsas entau compte d'un utilizator. OAuth qu'implica d'obtiéner un « geton d'accès », qui lo client utiliza entà autentificar las requèstas cap a un servidor de ressorsas.

La fòrça d'OAuth qu'ei la soa polivaléncia dens la gestion de diferents scenaris, de las aplicacions web aus aparelhs mobiles e a la comunicacion servidor-servidor. Que separa lo ròtle deu client (demanda d'accès) deu proprietari de la ressorsa (l'utilizator) e deu servidor d'autorizacion (validacion de l'utilizator e emission de getons).

Que consideratz ua aplicacion web qui utiliza OAuth entà accedir a las dadas d'un utilizator a partir d'un servici de réseautage sociau. Lo flux qu'implica tipicament :

1. Redirection de l'utilizator cap a la pagina d'autorizacion deu servici.
2. Dab lo son consentiment, lo servici que torna dirigir cap a l'aplicacion Web dab un còdi d'autorizacion.
3. L'aplicacion qu'escàmbia aqueth còdi contra un geton d'accès .
4. Fin finau, l'aplicacion qu'utiliza aqueth geton entà har aperets API au nom de l'utilizator.

Aqueth procediment que garanteish que los identificants de l'utilizator son pas expausats a l'aplicacion Web, melhorant atau la securitat.

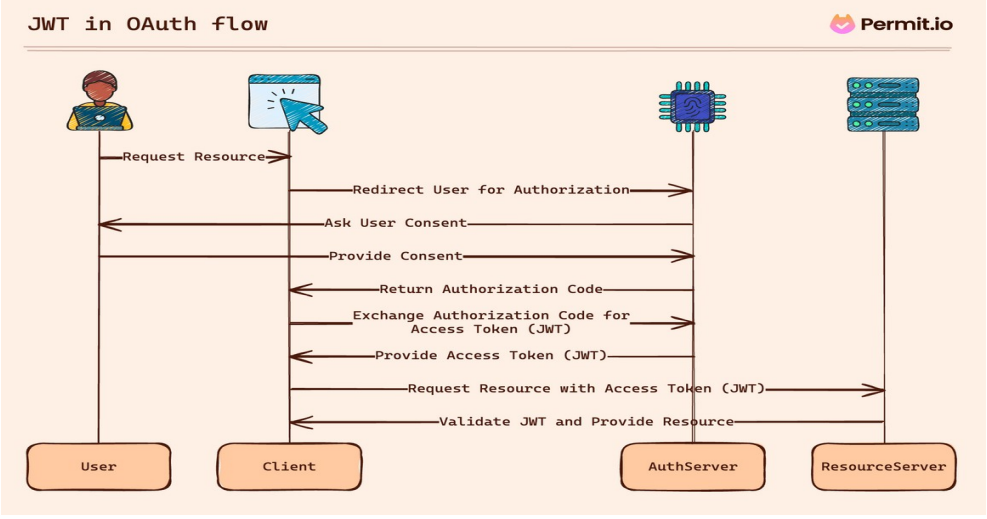
OAuth 2,0 en pregondor: Tipes de subvencions e flux

OAuth 2.0 que prepausa diferents tipes de subvencions entà diferents scenaris, cadun concebut entà hornir un accès delegat securizat dens condicions especificas.

1. **Garantida de còdi d'autorizacion** : Ideau entà las aplicacions costat servidor, aqueth tipe d'autrei qu'implica de tornar dirigir l'utilizator cap au servidor d'autorizacion, d'obtiéner un còdi d'autorizacion de l'utilizator e de l'escambiar contra un geton d'accès au servidor. Aqueth procediment qu'ahorteish la securitat en evitant l'exposicion dirècta deus getons a l'agent utilizator.
2. **Garantida Implicita** : Adaptat aus clients s'executant dens un navigator, aqueth tipe de subvencion que simplifica lo flux en tornant dirèctament un geton d'accès, en sautant l'estapa d'escambi de còdi d'autorizacion. Aquò que convien a las aplicacions qui pòden pas estocar en tota securitat lo secret deu client. Qu'ei aquí que los JWT luseishen, pr'amor que pòden estar verificats shens dependéncias de servidor dab l'ajuda de JWKs publics.
3. **Garantida d'identificants de mot de passa deu proprietari de ressorsa** : Ací, l'utilizator que horneish un nom d'utilizator e un mot de passa dirèctament a l'aplicacion, qui'us escàmbia puish contra un geton d'accès. Maugrat que simple, aqueth metòde qu'ei mensh segur e recomandada unicament entà las aplicacions hidablas.
4. **Garantida d'identificants clients** : Utilizat entà la comunicacion servidor-servidor, aqueth tipe de subvencion que permet au client d'obtiéner un geton d'accès dab l'ajuda deus sons identificants. Qu'ei ideau entà scenaris on l'aplicacion ageish entau son pròpi compte meilèu qu'entà lo compte d'un utilizator.

JWT e OAuth: Tribalhar amassa

Maugrat las lors diferéncias, JWT e OAuth que pòden tribalhar amassa eficaçament. Dens de nombrosas implementacions d'OAuth, los getons d'accès emetut que son en hèit JWT . Aquera combinason qu'espleita los punts hòrts de las duas tecnologias : lo quadre d'autorizacion robusta d'OAuth e la capacitat de JWT a encoder las informacions de l'utilizator e las reclamacions en tota securitat.



Per example, dens un flux OAuth 2.0, un còp qui lo client a recebut un geton d'accès (qui poiré estar un JWT), que pòt har requèstas au servidor de ressorsas au servidor de ressorsas. Lo servidor valide puish lo JWT, en s'assegurant qu'ei signat e non alterat, e que tira las informacions de l'utilizator e las autorizacions codadas a la soa popa.

Aquera combinason qu'ei particularament utila dens las aplicacions a pagina unica (SPA) e dens las aplicacions mobilas, on la natura apatrida de las JWT ajuda a mantiéner las sessions utilizator shens estocatge costat servidor, mentre qu'OAuth gereish las autorizacions e los camps de çò a qué lo client pòt accedir.

Consideracions relativas a la securitat

Maugrat que JWT e OAuth que hornissen tots dus de solides mecanismes de securitat, qu'ei essenciau de'us implementar corrèctament entà evitar las vulnerabilitats :

1. **Securizar JWT** : Qu'asseguratz que lo geton ei criptat e qu'utiliza un algoritme de signatura hòrt . Evitar d'expausar de las informacions sensiblas dens las cargas utilas JWT, pr'amor que pòden estar descodadas si son pas criptadas.
2. **Securitat OAuth** : Utilizar HTTPS entà totas las transaccions implicant OAuth. Siatz prudent dab la reencaminament deus URI e validatz-los entà empachar las atacas de redirection. Plan estocar e manipular los getons en tota securitat.

JWT e OAuth dens l'autorizacion

Un aute aspècte dens lo quau JWT e OAuth que jògan un ròtle important qu'ei l'autorizacion - l'estadi a lo quau ua demanda decideish de çò a qué un utilizator ei autorizat o privat d'accès. Dens los JWT, las reclamacions simbolicas que pòden hornir informacions importantas suus atributs de l'utilizator requerit entà las autorizacions. Dens los flux OAuth, los camps de getons que horneishen informacions suus camps d'aplicacion desirats deu geton especific.
Ua idèa faussa correnta tà çò de JWT e OAuth qu'ei la lor capacitat a èste utilizat com un servici complet d'autorizacion de demanda. Los desvolopaires que s'interròjan sovent suus camps d'aplicacions e las revindicacions dirèctament dens las aplicacions entà determinar si un utilizator pòt o ne pòt pas efectuar accions particularas. Aqueth apròchi que pòt condusir a de grèus problèmas:

- **Évolu**: entà cada politica navèra d'accès, lo geton o lo flux que deurà estar modificat e tornat validar.
- **Complexitat**: si ua politica càmbia per ròtle o reclamacion, lo còdi de la demanda que deu estar modificat entad ac sostiéner.
- **Au dessús de l'autorizacion**: los camps d'aplicacion e las reclamacions ne contienen pas que dadas entau subjècte (utilizator, aplicacion, etc.), mentre que las autorizacions son egaument motivadas per accions e de las ressorsas.

Entà relhevar aqueths desfís, los camps d'aplicacion e las reclamacions que deven estar utilizats dens un servici d'autorizacion qui descobla la politica deu còdi. Un servici d'autorizacion, tau com Permit.io, que deuré avalorar las decisions d'autorizacion en utilizant los camps e las reclamacions dens l'encastre d'un modèle d'autorizacion complet qui calcula tots los factors de securitat pertinents, vse permetent d'integrar las JWT e OAuth dens la vòsta aplicacion de faiçon transparenta.

Conclusion

Compréner OAuth e JWT qu'ei essenciau entau desvolopament modèrne deu web . Mentre qu'OAuth horneish un quadre d'autorizacion flexibla, JWT qu'auhereish un mejan compacte de representar las informacions de l'utilizator en tota securitat. Combinats, que forman ua combinason poderosa entà securizar las aplicacions web, hornint ua autentificacion hòrta e un contròle d'accès a grans fins.