esprit
Se former autrement
HONORIS UNITED UNIVERSITIES

JWT

JSON Web Tokens

Module Services Web

A.U 2025-2026

United Nations Educational, Scientific and Cultural Organization
UNESCO Chair "Project-based learning" ESPRIT School of engineering, Tunisia
EUR-ACE® Délivrée par la Commission des Titres d'Ingénieur
CONFÉRENCE DES GRANDES ÉCOLES
cdio CONCEIVE DESIGN IMPLEMENT OPERATE

Définition

- JSON Web Token (JWT) est un standard,
- Définit une solution, compacte et autonome,
- Permet de transmettre de manière sécurisée des informations entre les applications en tant qu'objet structuré au format JSON.

Compact ?

- En raison de leur ***petite taille***, les JWT peuvent être envoyés via une URL, un paramètre POST ou dans un en-tête HTTP.

- De plus, la plus petite taille signifie que la transmission est **rapide**.

Autonome?

- Le JWT contient toutes les informations requises sur l'utilisateur,

- Ce qui évite d'avoir à interroger la base de données plus d'une fois pour connaitre le détail de l'identité d'un client authentifié.

Structure de JWT?

- JWT est constitué de trois parties séparées par un point « . » :
  - **Header**
  - **Payload**
  - **Signature**

- La forme d'un JWT est donc : **xxx.yyy.zzz**

JWT : Header

- L'en-tête se compose généralement de deux parties:
  - **Le type du jeton**, qui est JWT,
  - **L'algorithme de hachage** utilisé, tel que : **HMAC (***HS512, HS256, HS384***)** ou **RSA**
- La structure du Header est un objet JSON ayant la forme la suivante :

```
{ "alg": "HS256",
  "typ": "JWT"
}
```

- Cet objet JSON est ensuite encodé en **Base64URL**.

JWT : Payload

- C'est la deuxième partie du jeton,
- Elle contient les claims suivants:
  - ***iss*** (issuer : Origine du token),
  - ***exp*** (heure d'expiration),
  - ***sub*** (sujet),
  - ***aud*** (public cible),
  - ***nbf*** (Not Before : A ne pas utiliser avant cette date) ,
  - ***iat*** (issued at : date de création du token),
  - ***jti*** (JWT ID identifiant unique du JWT).

JWT : Exemple de Payload

```
{
        "sub": "Ines",
        "iat":49865432,
        "exp":54789005,
        "nbf":null,  "jti":"idr56543ftu8909876",
        "roles":["admin","author"]
}
```

- Ce **payload** est aussi encodé en **Base64URL**.

JWT : Signature

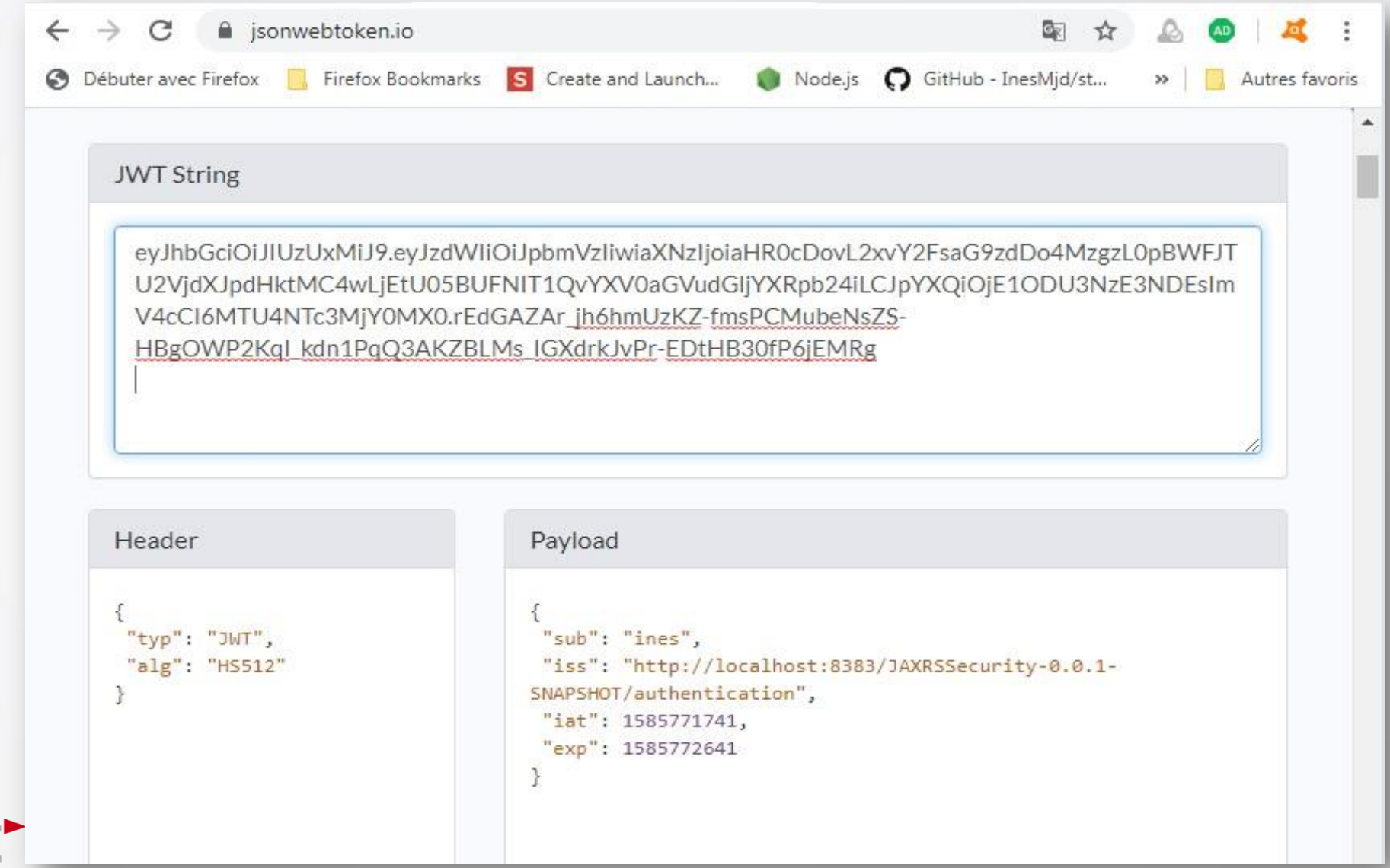
- C'est la dernière partie du jeton,
- Elle est utilisée pour:
  - vérifier que l'expéditeur du JWT est celui qu'il prétend être.
  - et pour s'assurer que le message n'a pas été modifié en cours de route.
- Si vous voulez utiliser l'algorithme **HMAC SHA256**, la signature sera créée de la façon suivante:
  - **HMACSHA256**(base64UrlEncode(**header**) + "**.**" + base64UrlEncode(**payload**), secret)

JWT

- Le JWT final est constitué des trois chaînes **Base64** séparées par **des points** qui peuvent être facilement transmis tout en étant plus compacts.

- L'exemple suivant montre un JWT qui a été signé avec un secret.

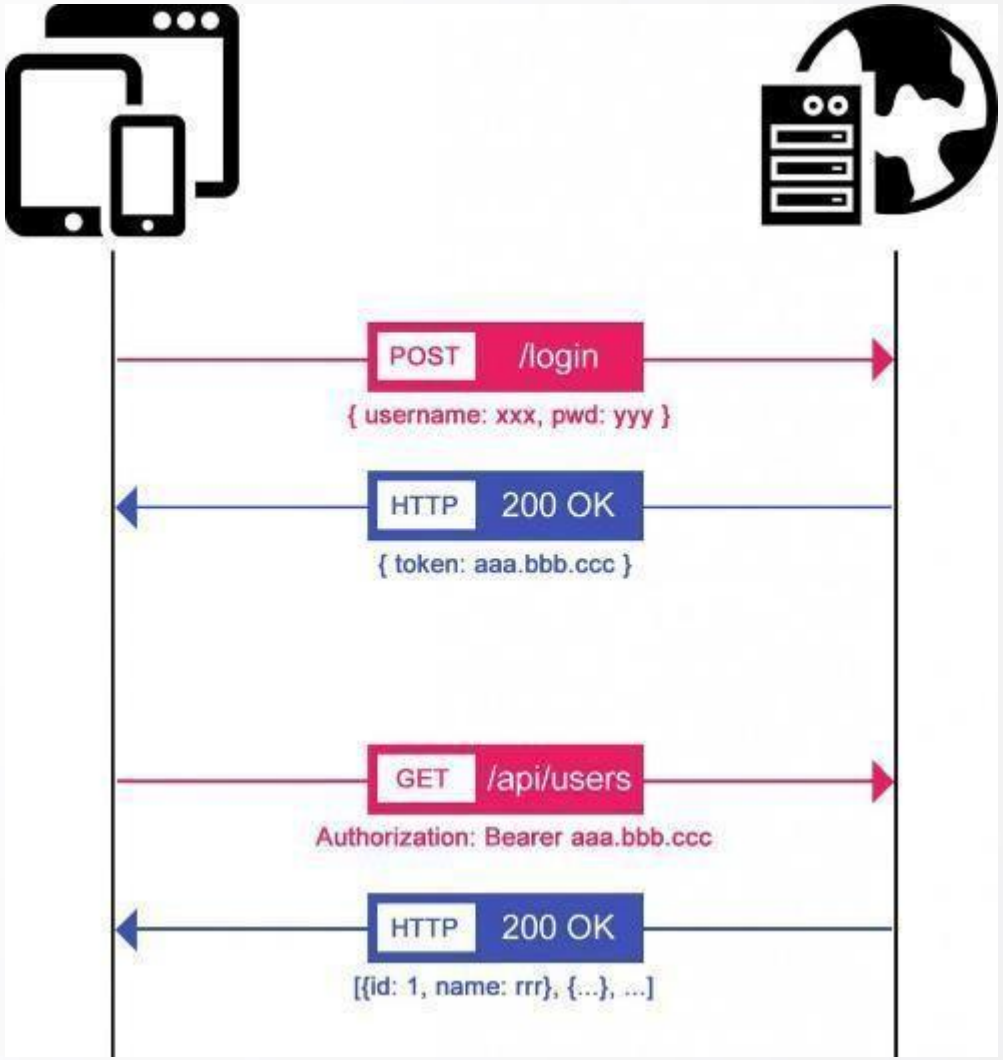
JWT: Exemple de JWT



Comment utiliser JWT

- **Au moment d'authentification:**
  - L'utilisateur se connecte avec ses informations d'identification,
  - Un JWT est renvoyé, il est enregistré généralement dans le storage Local.
- **Disposant de ce jeton,** le client doit maintenant envoyer une requêtes pour chaque accès à une ressources sécurisée:
  - Il doit envoyer le jeton généralement dans l'en-tête **Authorization** avec le schéma Bearer:

Authorization: Bearer <jeton>



JWT