Les services Web échangent des données entre eux par HTTP. Or, ce protocole n’est pas suffisamment sécurisé. WS-Security permet de pallier ce manque de sécurité.

WS-Security complète la spécification existante des messages en ajoutant des éléments dans les headers pour ajouter des mécanismes d’authentification, de signature ou encore de chiffrement.

* Authentification

Ce sont les tokens qui contiennent les identifiants de l’auteur du message. Typiquement, ces tokens contiennent le login et mot de passe. Le service recevant le token peut ensuite vérifier si les identifiants fournis sont les bons.

Les tokens principalement utilisés sont :

* + User Name Token (Couple login/Mot de passe)
  + SAML Token (Données d’authentification au format XML SAML)
  + X509 Token (Données d’authentification par certificat)

Il existe d’autres tokens utilisant d’autres systèmes, comme par exemple Kerberos.

Un header contenant un token ressemble à ceci :



* Chiffrement

Le chiffrement permet d’empêcher la lecture du message si celui-ci est intercepté.

La méthode de chiffrement utilisée peut être communiquée au recevant via la norme WS-Security. La norme permet également l’échange de clé si besoin.

Voici un exemple de message chiffré avec un certificat X509 :



* Signature

La signature permet de garantir l’origine d’un message.

Pour cela, un jeton chiffré est ajouté au message, dont le secret est partagé entre l’émetteur et le recevant. Le recevant peut donc vérifier le jeton, et vérifier ainsi l’origine du messsage.

Exemple :



Sources :

<https://www.guru99.com/security-web-services.html>

<http://www.journaldunet.com/developpeur/tutoriel/xml/070502-xml-ws-security.shtml>

<https://blog.octo.com/securite-des-services-web-1ere-partie/>