ATTAQUE XSS – Cross-Site Scripting

Carine GUIOLET

Qu’est-ce qu’une attaque XSS ?

- Le Cross-site Scripting est une vulnérabilité particulièrement répandue dans les applications web.
- Cette vulnérabilité de sécurité permet à un attaquant d’injecter du code malveillant (généralement du JavaScript) dans une page web consultée par d’autres utilisateurs.
- Cette faille est exploitée pour voler des informations sensibles, manipuler des sessions utilisateur ou rediriger des utilisateurs vers des sites malveillants.

Définition et fonctionnement d'une attaque XSS

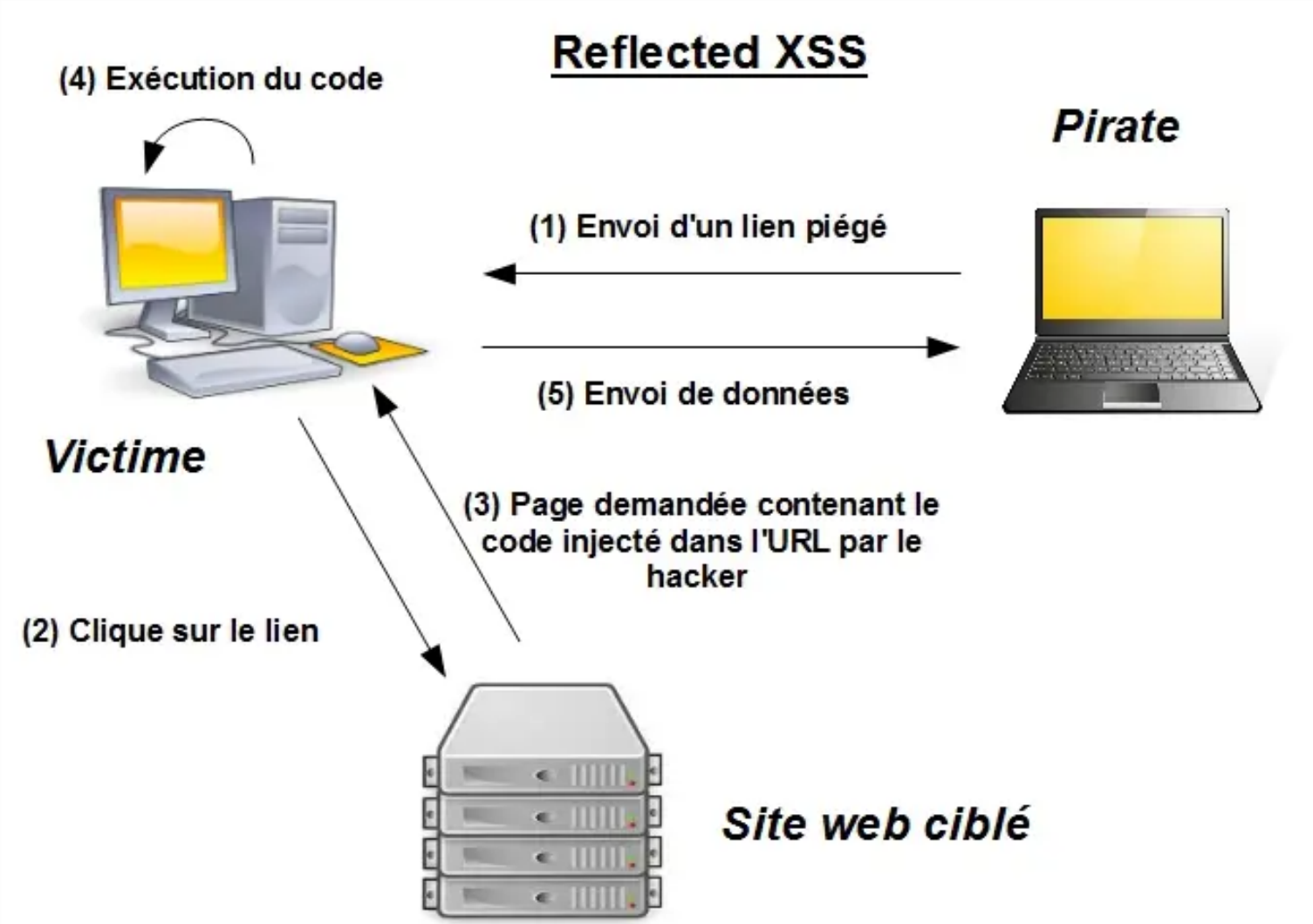
- Les XSS font partie de la catégorie des vulnérabilités par injection de code au même titre que les injections SQL.
- Les attaques XSS se produisent lorsque des applications web ne filtrent pas correctement les entrées utilisateur et affichent ces entrées de manière non sécurisée dans le navigateur.
- L'attaquant peut ainsi injecter des scripts malveillants qui seront exécutés par le navigateur des victimes.

Les différents types d’attaques XSS

- XSS Réfléchi (Reflected XSS)
  - Le code malveillant est injecté via une requête HTTP (ex : formulaire, URL)
  - Il est exécuté lorsque la victime clique sur lien piégé ou soumet un formulaire contenant du code malveillant.

- **Reflected XSS** (*non persistante*) :

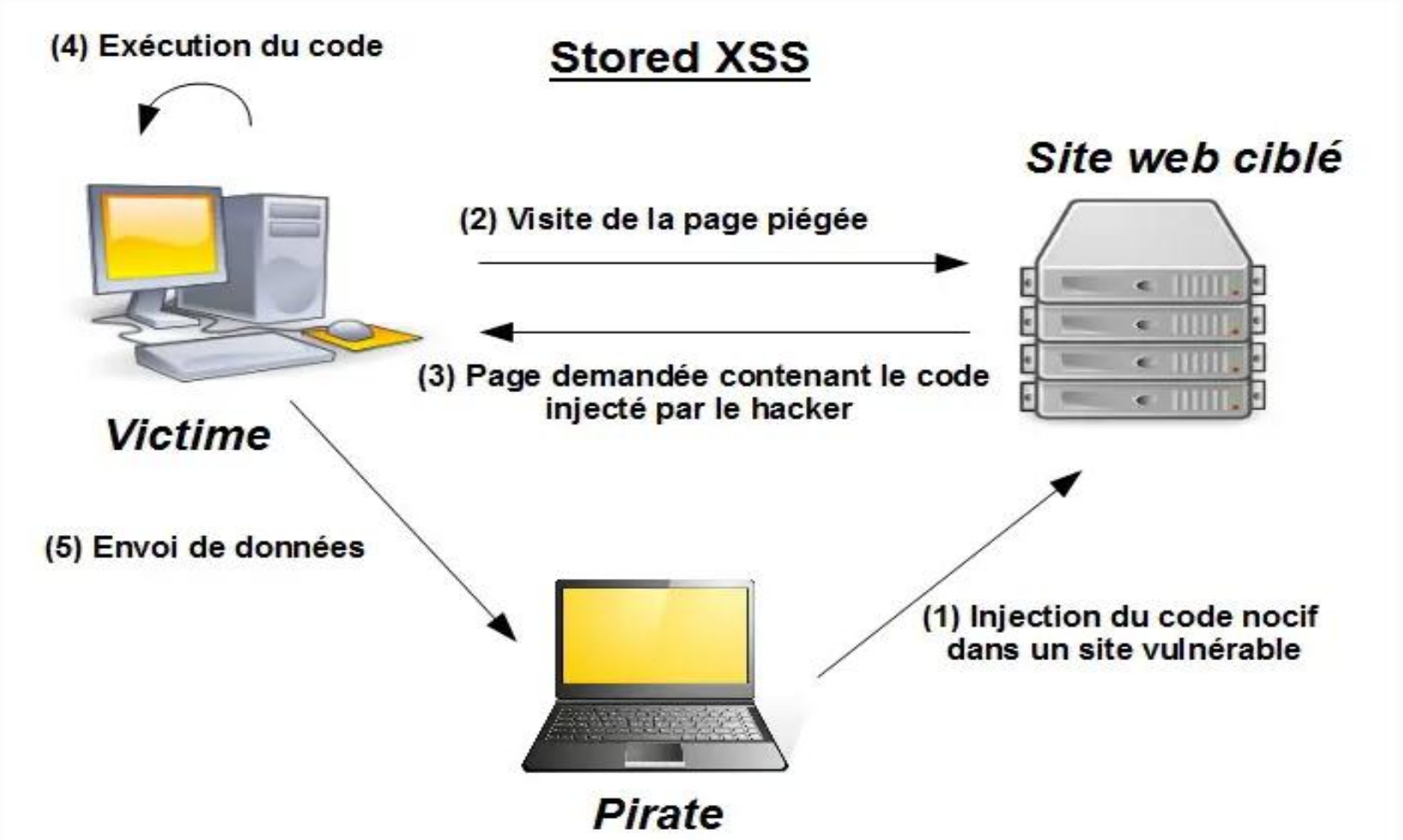
Lorsque des données sont envoyées par un client et sont affichées telles quelles dans la page résultante sans être encodées en entités HTML.



- XSS Persistant (Stored XSS)
  - L’attaquant injecte du code malveillant dans une base de données
  - Le script malveillant est ensuite affiché aux utilisateurs légitimes lorsqu'ils consultent une page web contenant les données stockées

- **Stored XSS** (*persistante*) :

Lorsque des données sont fournies depuis une source de données quelconque (BDD, fichiers, etc.) et sont affichées telles quelles dans la page résultante sans être encodées en entités HTML. L'impact d'une XSS stockée est d'autant plus grave car elle touche tous les visiteurs de la page piégée.



- XSS DOM (Document Object Model)
  - L’attaque se produit lorsque du JavaScript manipule le DOM de la page sans une validation adéquate des entrées utilisateur.
  - L’attaquant peut modifier dynamiquement le contenu affiché par la page.

Conséquences d’une attaque XSS

- Vol de cookies de session permettant de prendre le contrôle de comptes utilisateurs.
- Phishing et redirection vers des sites malveillants.
- Modification totale de site Web (un site qui devient méconnaissable).
- Exécution de commandes malveillantes sur le navigateur de la victime.

Comment se protéger d’une attaque XSS

Validation et filtrage des entrées utilisateur : Vérifier et encoder toutes les données saisies par l'utilisateur.

Utilisation d'headers HTTP sécurisés : Content Security Policy (CSP) pour limiter l'exécution des scripts.

Éviter l'injection directe dans le DOM : Privilégier les API sécurisées pour manipuler le DOM.

- Sources
  - https://www.clever-age.com/owasp-cross-site-scripting-xss/
  - https://owasp.org/www-project-top-ten/