Exploitation Web

Guillaume Pillot

Club de Hacking de l'Université Laval

2 Novembre 2016

Sommaire

Qu'est ce que le Web ?

- L'étape de scan nous a permis d'identifier les machines et les services disponibles dans le/les réseau(x) de notre cible dont les serveurs Web
- Le Web pour World Wide Web (WWW) consiste à naviguer entre des documents (page Web) reliés par des hyperliens
- Aujourd'hui, tout est connecté à internet (laptop, cellulaires, IoT, etc.) et le Web constitue un vecteur d'attaque important !
- Documentation : Lien 1 Lien 2 Lien 3
  Les bases du hacking. Chapitre 6 : Exploitation web p. 149 à 150

Qu'est ce qu'une URL ?

- Une URL (Uniform Resource Locator) est une adresse web permettant d'identifier de manière unique une ressource sur le web et d'y accéder
- Les hyperliens pointent donc vers des URLs
- Voici une URL permettant d'accéder à la ressource **scans.pdf** : http ://hacking.fsg.ulaval.ca/pdf/scans.pdf
- Documentation : Lien

Le HTML

- Les pages web sont écrite en HTML (HyperText Markup Language)
- C'est un langage de balisage qui structure le contenu des pages :

```
<!-- Exemple de base d'une page HTML -->
<!-- DOCTYPE pour le HTML5 -->
<!DOCTYPE html>
<html>
 <head>
 <!-- lien vers notre feuille de style -->
 <link rel="stylesheet" href="rouge.css" />
  <title>
   Mon titre!
  </title>
 </head>
 <body>
  <p>
   Blabla!
  </p>
<!-- chargement de notre script alert.js -->
 <script src="alert.js"></script>
 </body>
</html>
```

- Documentation : Lien

Le CSS

- Le CSS (Cascading Style Sheets) ou feuille de style est un autre langage qui permet de mettre en forme la page :

```
/* règle CSS qui met tout contenu dans une balise p en rouge */
p{
        color : red;
}
```

- Documentation : Lien

Le navigateur web

- Un navigateur web permet d'interprété le HTML et le CSS
- Il existe plusieurs navigateurs, les plus connus sont :
  - Firefox : Le navigateur libre, nous recommandons celui-ci !
  - Google Chrome : Le navigateur de Google
  - Internet Explorer : Le navigateur de Microsoft
  - Safari : Le navigateur pour Mac
- Documentation : Lien

Le JavaScript

- Depuis le début du web, les technologies ont beaucoup évolué et il est aujourd'hui possible de dynamiser les pages web grâce au langage de script
- L'un des langages de script les plus connus et les plus utilisés aujourd'hui est le JavaScript
- JavaScript est un langage de script côté client, c'est-à-dire que les scripts sont exécutés par le navigateur web de l'utilisateur et non pas par le serveur web

Le JavaScript

- JavaScript est un langage très puissant et permet de réaliser énormément d'action comme la gestion des événements (clique, survol de la souris, etc.), formulaire interactif, AJAX, etc.

```
//Affiche une boîte de dialogue "COUCOU!"
alert('COUCOU!');
/*
On récupère toute balise p de la page et on les place dans un tableau
Dans notre exemple le tableau fait une case
Puisqu'il n'y a qu'une seule balise p dans notre page
*/
var elem = document.getElementsByTagName('p');
//On ajoute ensuite ce code HTML dans la page
elem[0].innerHTML += '<strong>Bloblo!</strong>';
```

- Documentation : Lien

Débuggage

- Lorsque l'on veut tester le fonctionnement d'une page web, la première action à faire est d'afficher le code source de la page
- Avec Firefox, il suffit d'utiliser le raccourci **Ctrl+U**
- Ensuite, la plupart des navigateurs possèdent un débogueur permettant d'analyser et de modifier le comportement et le contenu de la page
- Dans cet atelier, nous utiliserons le plugin Firebug
- Documentation : Lien

Challenge Time

- Challenge 1
- Challenge 2
- Challenge 3

Système binaire

- Un ordinateur ne fait qu'exécuter une suite de bits. Un bit valant soit 1, soit 0
- Ce système de numérotation se nomme système binaire
- L'unité d'informations utilisée en informatique est l'octet composé de 8 bits
- Un octet permet entre autres de stocker un caractère
- Voici un octet **01001101**, pour le convertir en un nombre décimal, il faut effectuer l'opération suivante :

$$N = 0*2^7+1*2^6+0*2^5+0*2^4+1*2^3+1*2^2+0*2^1+1*2^0 \quad (1)$$

Ce qui nous donne **77**

- Documentation : Lien

Système hexadécimal

- Un nombre binaire n'ayant que deux symboles peut être très long. Donc pour en faciliter la lecture, le système hexadécimal est utilisé
- Le système hexadécimal contient 16 symboles. Les chiffres 0 à 9 et les lettres de A à F
- Pour convertir notre octet **01001101** en un nombre hexadécimal, il faut diviser les bits par paquet de 4 et effectuer les opérations suivantes :

$$H1 = 0 * 2^3 + 1 * 2^2 + 0 * 2^1 + 0 * 2^0 = 4_{10} = 4_{16} \quad (2)$$

$$H2 = 1 * 2^3 + 1 * 2^2 + 0 * 2^1 + 1 * 2^0 = 13_{10} = D_{16} \quad (3)$$

  Ce qui nous donne **4D**
- Documentation : Lien

ASCII

- L'ASCII (American Standard Code for Information Interchange) est une norme de codage de caractères en informatique
- Le principe est simple, chaque lettre de l'alphabet correspond à une suite de bits
- Il existe 127 caractère dans cette norme, on a donc seulement besoin de 7 bits pour encoder un caractères, le bit de poids fort étant à 0
- Pour voir à quelle suite de bit correspond un caractère, il suffit de consulter une table ASCII
- Par exemple, notre octet **4D** correspond à la lettre M majuscule
- Documentation : Lien

Encodage URL

- Les applications web utilisent plusieurs encodages différents pour leur donnée, l'une d'entre elles est l'encodage URL
- Les URLs ne peuvent contenir que des caractères imprimables, soit les caractères **20** à **7E** dans la table ASCII
- L'encodage URL permet d'encoder les caractères ayant une signification spéciale (comme le caractère + qui signifie espace dans une URL) pour que ceci ne soit pas interprété

Encodage URL

- Un encodage URL est préfixé du caractère % suivi de son code ASCII hexadécimal :
  - %3d vaut =
  - %25 vaut %
  - %20 vaut espace
- L'encodage URL de la chaîne de caractère **0+0= la tete a toto** vaut :
  %30%2b%30%3d%20%6c%61%20%74%65%74%65%20%61%20%74%6f%74%6f
- Documentation : The Web Application Hacker's Handbook. Chapitre 3 : Web Application Technologies p. 67

Encodage HTML

- Certains caractères spéciaux comme le ', le " ou les chevrons ($<$ et $>$) ne peuvent être interprété comme du contenu tel quel
- L'encodage HTML permet d'encoder ces caractères pour qu'il soit considéré comme du contenu

Encodage HTML

- Un encodage HTML est préfixé du caractère & suivi, soit d'une chaîne de caractères spéciaux :
  - &quot ; pour "
  - &apos ; pour '
  - &amp ; pour &

  Soit de # et du code ASCII décimal du caractère :
  - &#34 ; pour "
  - &#39 pour '

  Soit de # et du code ASCII hexadécimal du caractère :
  - &#x22 ; pour "
  - &#x27 ; pour '
- L'encodage HTML de la chaîne **Bob's&Co** vaut :
  &#x42 ;&#x6f ;&#x62 ;&#x27 ;&#x73 ;&#x26 ;&#x43 ;&#x6f ;
- Documentation : The Web Application Hacker's Handbook. Chapitre 3 : Web Application Technologies p. 68-69

Encodage en base64

- L'encodage en base64 permet de représenter n'importe quelle donnée binaire avec seulement des caractères ASCII imprimables. On l'utilise principalement pour la transmission de message sur internet (comme les courriels) étant donné que la plupart des protocoles n'acceptent que les caractères imprimables
- Cet encodage utilise les 64 caractères suivants :
  ABCDEFGHIJKLMNOPQRSTUVWXYZ
  abcdefghijklmnopqrstuvwxyz
  0123456789+/

Encodage en base64

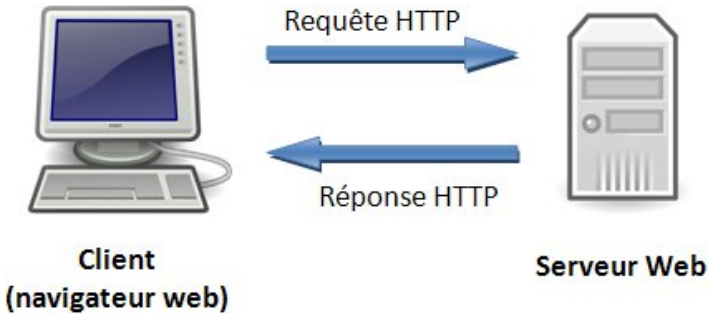
- Pour encoder la chaîne **Thee**, il faut d'abord sa valeur binaire que l'on va découper en groupe de 6 :
  010101|000110|100001|100101|011001|010000
  Si le dernier groupe n'a pas une taille de 6 bits, on rajoute des zéros pour qu'il le soit, nous avons donc rajouté 4 zéros
- Ensuite, on convertit ces nombres binaires en valeur décimale :
  21|6|33|37|25|16
- Ses chiffres correspondent aux index des caractères en base64, nous avons donc la chaîne base64 suivante : **VGhlZQ**
- La taille de la chaîne doit être un multiple de 4, si ce n'est pas le cas, on rajoute un nombre suffisant de caractères = à la fin de la chaîne, nous avons donc finalement : **VGhlZQ==**
- Documentation : Lien
  The Web Application Hacker's Handbook. Chapitre 3 : Web Application Technologies p. 69

Challenge time

- Challenge 1
- Challenge 2
- Challenge 3
- Challenge 4

Le protocole HTTP

- Le protocole HTTP (HyperText Transfer Protocol) est un protocole de la couche application
- Son but est de permettre le transfert de fichier localisé par une URL
- Le client (le navigateur web) envoie une requête HTTP au serveur web. Celui-ci traite la requête et envoie une réponse HTTP



- Il existe plusieurs serveurs web, les plus connus étant Apache et Nginx
- Documentation : Lien
  The Web Application Hacker's Handbook. Chapitre 3 : Web Application Technologies p. 39-40

Requête HTTP

- Les messages HTTP (requête et réponse) constituent en un ou plusieurs entêtes séparé entre elles par une ligne, suivi obligatoirement par une ligne blanche
- Une requête HTTP débute par une ligne constituée de la méthode HTTP utilisée, suivi de l'URL ciblé et finissant par la version HTTP utilisée
- Ensuite, les autres lignes représentent les différents entêtes respectant le format suivant : **`nom-header: value`**

Requête HTTP

- Voici un exemple de requête HTTP :

```
GET / HTTP/1.1
Host: hacking.fsg.ulaval.ca
User-Agent: Mozilla/5.0 Gecko/20100101 Firefox/45.0
Accept: text/html,application/xhtml+xml,application/xml
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
```

- Documentation : Lien
  The Web Application Hacker's Handbook. Chapitre 3 : Web Application Technologies p. 40-41

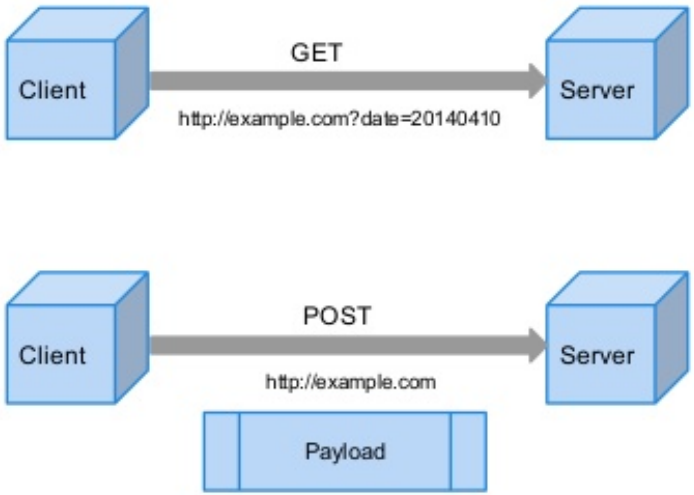
Méthode HTTP

- Il existe plusieurs méthodes HTTP, les plus courantes étant les méthodes GET et POST
- La méthode GET permet de simplement demander une ressource. Il est possible d'envoyer de données directement dans l'URL comme ceci :

  http ://www.exemple.com/page.html ?couleur=bleu&forme=rectangle

  Ici, nous envoyons la donnée couleur avec la valeur bleu ainsi que la donnée forme avec la valeur rectangle
- La méthode POST envoie les données (payload) dans la requête HTTP à la suite des entêtes

Méthode HTTP



- La méthode GET devrait être utilisée lorsque la requête ne modifie pas les données. POST est utilisée pour l'envoi de donnée complexe (upload multimédia) et pour la validation de formulaire
- Documentation : Lien

Réponse HTTP

- Une réponse HTTP contient en première ligne la version HTTP utilisée, suivi du code du résultat du statut de la requête et suivi de la signification de ce code
- Ensuite, la réponse contient plusieurs entêtes (du même format que pour les requêtes), suivi du corps de la réponse (HTML ou autres)

Réponse HTTP

- Voici un exemple de réponse HTTP :

```
HTTP/1.1 200 OK
Date: Tue, 25 Oct 2016 20:01:58 GMT
Server: Apache/2.2.14 (Ubuntu)
X-Powered-By: PHP/5.3.2-1ubuntu4.30
Content-Type: text/html

<!doctype html>
<html>
<head>
<!-- Reste de la page HTML -->
```

Réponse HTTP

- Il existe plusieurs code différents dont voici les plus connus :

Code	Signification	Description
200	OK	La requête a été accomplie
301	MOVED	Il s'agit d'une redirection, les données sont situé dans une nouvelle adresse
404	NOT FOUND	Le serveur n'a pas trouvé les données demandées
500	INTERNAL ERROR	Le serveur a rencontré une erreur durant le traitement de la requête

- Documentation : Lien
  The Web Application Hacker's Handbook. Chapitre 3 : Web Application Technologies p. 48-49

Cookie

- Un cookie est une suite d'informations généré par un serveur web pour un de ces clients que celui-ci conserve dans son disque dur, lui permettant entre autres de s'authentifier
- Le serveur fournit le cookie dans une réponse HTTP via l'entête `Set-Cookie` et le client le lui renvoi dans une requête via l'entête `Cookie`
- On appelle Cookie de session celui qui permet au site web de vous reconnaître lors de votre visite (par exemple, pour un site de e-commerce, il permet de conserver votre panier d'achat)
- Subtiliser un cookie de session et l'utiliser dans nos requêtes HTTP vers le serveur web (en remplaçant la valeur de l'entête `Cookie`), nous permet donc d'usurper l'identité d'un utilisateur sur le site du serveur
- Documentation : Lien 1 Lien 2
  The Web Application Hacker's Handbook, Chapitre 3 : Web

Installation et configuration

- Burp Suite est un logiciel permettant de tester la sécurité des applications web
- Il est installé par défaut sur Kali Linux, mais il faut configurer son navigateur web en suivant les instructions suivantes et installer son certificat SSL (Secure Sockets Layer) pour intercepter les requêtes HTTPS
- HTTPS (S pour Secure) est la combinaison du HTTP avec une couche de chiffrement comme SSL permettant de crypter la communication
- Ce protocole utilise des certificats électroniques signés par une autorité de certification (CA) pour valider l'identité du serveur web
- Burp Suite n'étant pas reconnu par une autorité de certification, il faut installer son certificat SSL
- Documentation : Lien

Burp Proxy

- Un proxy se situe entre le navigateur et le serveur web. Il permet d'intercepter les requêtes HTTP
- Le proxy de Burp Suite est son outil principal. En plus d'intercepter les requêtes HTTP, il nous permet de les éditer
- Documentation : Lien

Challenge Time

- Challenge 1
- Challenge 2
- Challenge 3

Burp Spider

- Burp Spider permet de mapper une application web
- Il va ajouter toute ressource pointée par un hyperlien dans la page intercepté
- Documentation : Lien

Challenge Time

- Utilisez Burp Spider sur https ://ringzer0team.com/ pour résoudre le challenge !
- Challenge

Burp Intruder

- Burp Intruder permet d'automatiser des attaques personnalisées contre une application web
- Cet outil est très complet et nous n'en verrons qu'une utilisation de base
- Nous allons vérifier le nombre de challenges disponible sur ringzer0 en utilisant un payload de type number
- Documentation : Lien

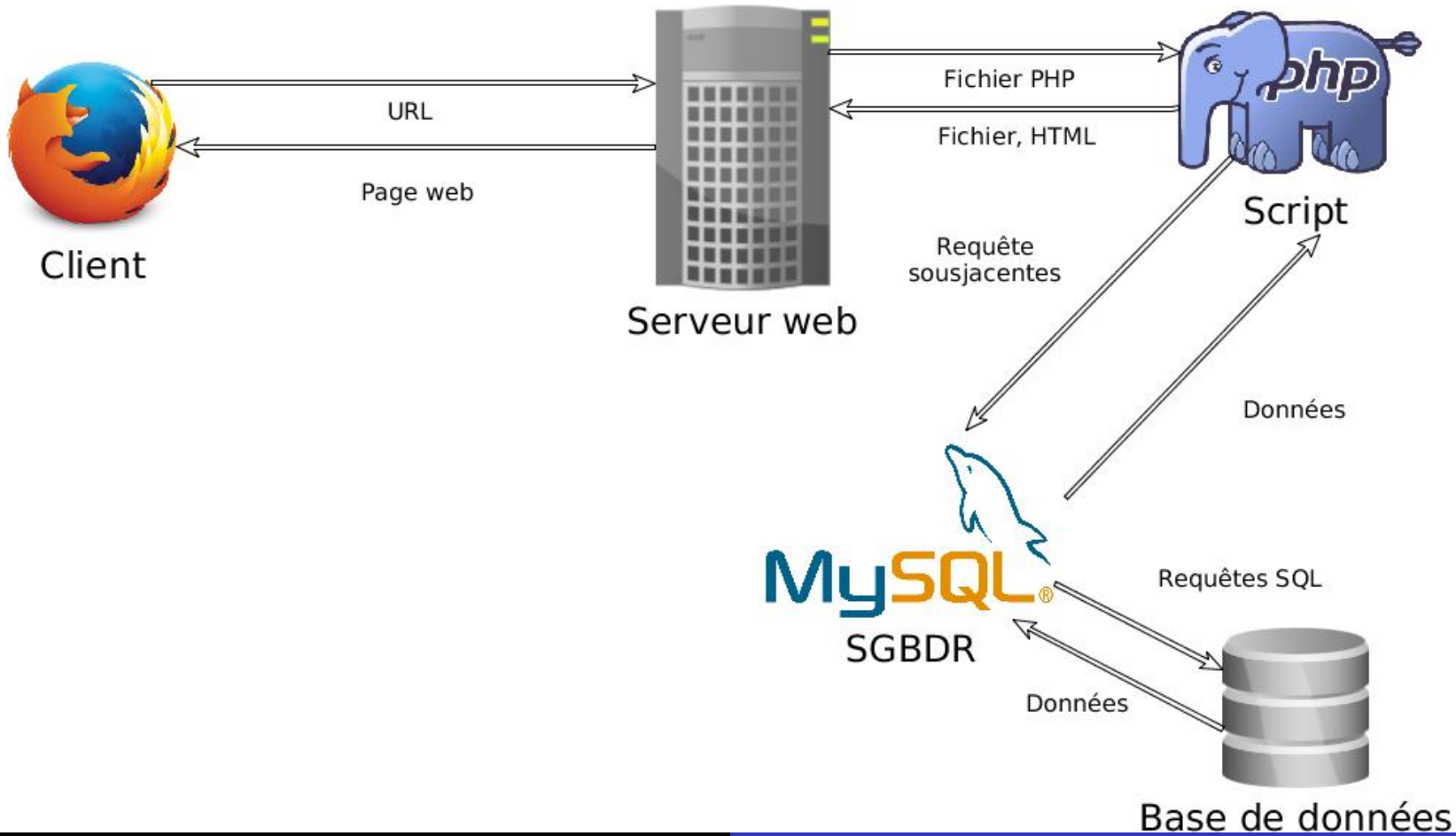
Challenge Time

- Avec Burp Intruder, trouvez l'ID existant vers l'URL http ://10.10.20.104 :31337/ID (où il faut remplacer ID par un nombre)

Langage Web et SGBDR

- Aujourd'hui, la grande majorité des sites web propose du contenu dynamique et une interaction avec leur utilisateur ce qui nécessite l'utilisation d'un langage web pour générer les pages
- Les langages les plus populaires en ce moment sont le PHP, ASP.NET, J2EE, Python avec le framework Django et Ruby avec le framework ROR
- Pour stocker et traiter les données des utilisateurs, un SGBDR (Système de gestion de base de données) tel MySQL, PostGreSQL ou Oracle doit être utilisée
- Documentation : Lien 1 Lien 2 Lien 3 Lien 4 Lien 5

Langage Web et SGBDR



Le langage SQL

- Le langage utilisé par le SGBDR est le SQL (Structured Query Language) qui permet d'exploiter des bases de données relationnelles
- Dans une base de données relationnelle, les données sont contenues dans des tables composées d'un entête et d'un corps
- Par exemple, nous avons la table Users suivante :

ID	nom	password	role
1	admin	r0xx0rz	admin
2	Dupont	tintin	user
3	Tremblay	Defroid	user
4	sushi	spyvscat5	hacker

Le langage SQL

- Les requêtes SQL permettent de manipuler une base de données
- Par exemple, la requête suivante permet d'obtenir le nom et le rôle de l'utilisateur admin :

```
SELECT nom,role FROM Users WHERE ID=1 ;
```

- Documentation : Lien

Injection SQL de base

- Pour s'authentifier, l'utilisateur rentre, via un formulaire, son nom et son mot de passe, nous avons donc la requête SQL suivante :

```
SELECT role FROM Users
WHERE nom=$input_nom AND password=$input_password ;
```

$input_nom et $input_password représentent ce que l'utilisateur a envoyé dans le formulaire

- Si l'utilisateur rentre Dupont avec le password tintin, la requête suivante sera exécutée :

```
SELECT role FROM Users
WHERE nom='Dupont' AND password='Tintin' ;
```

L'apostrophe indique le début et la fin d'une chaîne de caractère

Injection SQL de base

- Si l'utilisateur rentre un apostrophe dans l'un des input, le caractère sera interprété par le SGBDR et cela provoquera une erreur :

```
# Erreur de syntaxe
# Avec un apostrophe en trop à password
SELECT role FROM Users
WHERE nom='Dupont' AND password='''';
```

Injection SQL de base

- Si l'utilisateur rentre l'input suivant `admin'#`, la requête sera exécutée sans que l'utilisateur ait besoin de connaître le mot de passe de l'admin :

```
# Le dièse commente le reste de la requête
# Et permet d'éviter l'erreur SQL
SELECT role FROM Users
WHERE nom='admin'#' AND password='random';
```

- Documentation : Les bases du hacking. Chapitre 6 : Exploitation web p. 149 à 150
  The Web Application Hacker's Handbook. Chapitre 9 : Attacking Data Stores p. 287

Challenge Time

- Challenge 1
- Challenge 2

- Plusieurs méthodes existent, la plus efficace étant d'utiliser des requêtes préparées permettant que tout input ne puisse pas être exécutée dans une requête
- Documentation : The Web Application Hacker's Handbook. Chapitre 9 : Attacking Data Stores p. 338

Principe de base

- Le XSS (Cross-Site Scripting) est une des failles les plus connues. Elle a toujours été dans l'OWASP top 10
- Elle consiste à injecter des balises HTML afin de pouvoir exécuter du code d'un langage de script comme Java Script, la plupart du temps via un formulaire
- Lors d'un envoi de donnée par formulaire, par exemple, le post d'un forum, si le code n'est pas protégé et que l'utilisateur rentre le code ci-dessous, il sera exécuté par tout utilisateur visitant la page :

```
<script>
alert('XSS');
</script>
```

- Documentation : Lien
  Les bases du hacking. Chapitre 6 : Exploitation web p. 166 à 168
  The Web Application Hacker's Handbook. Chapitre 12 : Attacking Users : Cross-Site Scriting p. 431

Challenge Time

- Pour réussir ce challenge, vous devez usurper le cookie de l'administrateur (un bot qui visite régulièrement la page)
- Il faudra rediriger le bot vers votre serveur web et faire en sorte qu'il vous envoie son cookie
- JavaScript permet de faire tout cela : (Lien 1 Lien 2 Lien 3)
- Vous pouvez utiliser http ://requestb.in/ pour rediriger l'administrateur
- Challenge

Comment s'en protéger ?

- On peut se protéger en utilisant une liste de caractères autorisée (whitelist) et utiliser des fonctions standards d'échappement de caractères
- L'utilisation d'une liste noire est dangereuse et inefficace, car on ne peut tout prévoir
- En PHP, on peut utiliser les fonctions htmlspecialchars ou htmlentities
- Documentation : The Web Application Hacker's Handbook. Chapitre 12 : Attacking Users : Cross-Site Scriting p. 492

Challenge Bonus

- Challenge réalisé par l'ETS, il y a plusieurs flags à trouver, les trois premiers sont à votre portée : ETS
- Challenge crée par Ian Bouchard pour le club, il y a trois flags à trouver : http ://159.203.173.168/