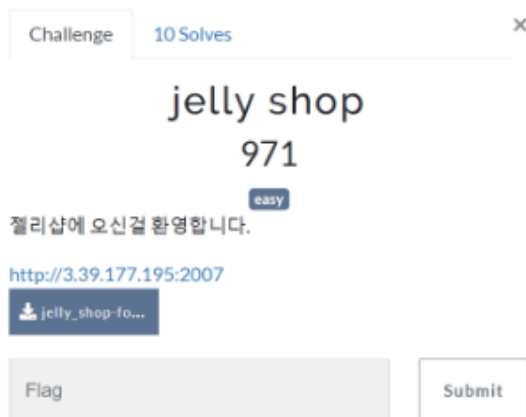


## 5월 Space CTF #8 Writeup

32기 안현진

이번 CTF에서는 'jelly shop'을 풀어보았다. Easy난이도이고, 문제 제목만 봤을 때 어떤 것을 다룰지 감이 잡히지 않아서 반대 더 흥미가 느껴져 이 문제를 고르게 되었다.


문제를 클릭하면 다음과 같은 화면이 보인다. 일단 밑에 첨부된 파일을 열어 힌트가 될 만한 정보를 찾아보도록하자.



압축을 푼 상태이다. 위에서부터 차례대로 열어가며 힌트를 찾아보자.

📁 .vscode	2024-05-11 오후 5:26	파일 폴더	
📁 __pycache__	2024-05-11 오후 5:26	파일 폴더	
📁 plag	2024-05-11 오후 5:26	파일 폴더	
📁 templates	2024-05-11 오후 5:26	파일 폴더	
📄 app.py	2024-05-11 오후 5:26	Python.File	1KB
📄 docker-compose.yml	2024-05-11 오후 5:26	YML 파일	1KB
📄 dockerfile	2024-05-11 오후 5:26	파일	1KB

1) vscode 파일을 열면 아래 그림과 같은 파일이 하나 뜬다.

이름	수정한 날짜	유형	크기
 settings	2024-05-11 오후 5:26	JSON File	1KB

처음 보는 파일의 형태라 **JSON File**에 대해 검색해봤다.

- JSON File은 “JavaScript Object Notation”의 약자로, 데이터를 저장하거나 전송하기 위한 경량의 형식이다.

일반적으로 텍스트를 기반으로 하는 파일 형식이라, 사람이 읽고 쓰기 쉽고 기계가 분석하고 생성하기 쉽다고 한다. 사람이 읽기 쉬운 형식이라면 어떤 힌트를 얻을 수 있지 않을까? 파일을 열어보자.

```

1  {
2    "iis.configDir": ""
3  }
```

이런 메시지가 나왔다. [“iis.configDir”: “”]에서 config와 Dir을 분리해서 보면 본적 있는 문장들이지만, 전체가 의미하는 바가 무엇인지는 아직 모르겠다.

”” 안에 있는 문장을 검색 해 보니, iis.configDir은 iis 즉 Microsoft Internet Information Services의 설정 디렉토리를 지정하는 것이라고 한다. iis는 윈도우를 사용하는 서버들을 위한 인터넷 기반 서비스들의 모임이다. 일단 여기서 뭘 얻어야 할지 모르겠으니 다음 파일을 열어보자

2)\_pycache\_을 열어봤지만, 프롬프트로 넘어가는 단계에서 계속 창이 없어졌다. Compiled Python File 유형이니 파이썬과 분명 관련이 있을 것이다. 뭐가 문제인지 몰라 아나콘다를 지웠다 깔고 파이썬과 관련된 앱, 정보를 계속 찾아보았지만 정말 해결이 되지 않아 일단 두번째 파일은 건너뛰고 진행했다(나중에 문제가 해결되면 다시 돌아오자)

3) plag는 텍스트 파일이다.

```
hspace{fakeflag}
```

fackflag.? 가짜 flag? 의미가 없는 파일이라고 생각하기도 했지만, 제작자가 정말 아

무 의도 없이 넣어놨을까? 일단 다음 파일로 넘어가 연결고리를 찾아보자.

4) templates은 HTML문서 이므로 Google Chrom으로 먼저 열어보고, 페이지를 어떻게 html로 작성했는지 알아보기 위해 text reader로 실행했다.

먼저 구글 크롬으로 실행하면 hspace war의 메인 화면에서 jelly shop을 클릭했을 때 보이는 주소와 비슷한 화면이 뜬다. 왼쪽 사진이 메인 화면에서 들어간 화면, 오른쪽 사진이 templates에 있던 파일 주소로 접속한 화면이다. 그리고 결제 화면은 이렇게 뜬다.

장바구니

#	상품	가격	수량	총액	
1	라임 젤리	\$25	0	\$0	삭제
2	오렌지 젤리	\$20	0	\$0	삭제
3	스트로베리 젤리	\$10	0	\$0	삭제
4	레몬 젤리	\$15	0	\$0	삭제

총액: \$0  
결제하기

장바구니

#	상품	가격	수량	총액	
1	라임 젤리	\$25	0	\$0	삭제
2	스트로베리 젤리	\$10	0	\$0	삭제
3	레몬 젤리	\$15	0	\$0	삭제
4	오렌지 젤리	\$20	0	\$0	삭제

총액: \$0  
결제하기

결제 확인

결제하시겠습니까?

주문 번호: #HSPACE-150536

결제 금액: \$15

취소 확인

오렌지 젤리 - 스트로베리 젤리 - 레몬 젤리 순서가 하나씩 밀려있다.

텍스트 리더로 열어보자. Html을 기반으로 하는(정확하지 않다. Html을 아직 배우지 않아 모르지만 간간히 for과 같은 C언어와 파이썬에서도 쓰이는 조건문이 보여서..) 복잡한 코드가 출력된다.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Document</title>
  <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/boots
trap.min.css" rel="stylesheet">
</head>
<body>
  <div class="container mt-5">
    <h1>장바구니</h1>
    <table class="table">
      <thead>
        <tr>
          <th scope="col">#</th>
          <th scope="col">상품</th>
          <th scope="col">가격</th>
          <th scope="col">수량</th>
          <th scope="col">총액</th>
          <th scope="col"></th>
        </tr>
      </thead>
      <tbody id="products">
      </tbody>
    </table>
    <div class="text-end">
      <h5>총액: $0</h5>
      <button class="btn btn-primary"
id="paymentButton">결제하기</button>
    </div>
  </div>
  <div class="modal fade" id="paymentModal" tabindex="-1" aria-
labelledby="paymentModalLabel" aria-hidden="true">
    <div class="modal-dialog modal-dialog-centered">
      <div class="modal-content rounded-3 border-0 shadow-lg">
        <div class="modal-header bg-primary text-white">
          <h5 class="modal-title fw-bold"
id="paymentModalLabel">결제 확인</h5>
          <button type="button" class="btn-close btn-close-white"
data-bs-dismiss="modal" aria-label="Close"></button>
        </div>
        <div class="modal-body" id="paymentModalBody">
```

```

        <div class="text-center mb-4">
            <i class="bi bi-check-circle-fill text-success fs-
3"></i>
        </div>
        <p class="fs-5 text-center mb-4">결제하시겠습니까?</p>
        <p class="text-center">주문 번호: <strong
id="orderNumber">#SPACE-</strong></p>
        <p class="text-center">결제 금액: <strong
id="paymentAmount">$100</strong></p>
    </div>
    <div class="modal-footer bg-light border-0">
        <button type="button" class="btn btn-secondary" data-bs-
dismiss="modal">취소</button>
        <button type="button" class="btn btn-primary"
id="paymentConfirm">확인</button>
    </div>
</div>
</div>
</div>
<div class="modal fade" id="paymentCompleteModal" tabindex="-1"
aria-labelledby="paymentCompleteModalLabel" aria-hidden="true">
    <div class="modal-dialog modal-dialog-centered">
        <div class="modal-content rounded-3 border-0 shadow-lg">
            <div class="modal-header bg-success text-white">
                <h5 class="modal-title fw-bold"
id="paymentCompleteModalLabel">결제 완료</h5>
                <button type="button" class="btn-close btn-close-white"
data-bs-dismiss="modal" aria-label="Close"></button>
            </div>
            <div class="modal-body">
                <div class="text-center mb-4">
                    <i class="bi bi-check-circle-fill text-success fs-
5"></i>
                </div>
                <p class="fs-5 text-center mb-4">결제가 성공적으로
완료되었습니다.</p>
                <p class="fs-5 text-center mb-4"
id="thanksalot">구매해주셔서 감사합니다.</p>
            </div>
            <div class="modal-footer bg-light border-0">
                <button type="button" class="btn btn-primary" data-bs-
dismiss="modal" id="paymentCompleteConfirm">확인</button>
            </div>
        </div>
    </div>
</div>
<script id="__PRODUCT_DATA__"
type="application/json">{"products":[{"title":"strawberry_jelly",
"display_name":"스트로베리

```

```

젤리“,“price“:10},{“title“:“lemon_jelly“,“display_name“:“레몬
젤리“,“price“:15},{“title“:“orange_jelly“,“display_name“:“오렌지
젤리“,“price“:20},{“title“:“lime_jelly“,“display_name“:“라임
젤리“,“price“:25}]]</script>
<script>
  document.addEventListener(“DOMContentLoaded“, async ()=>{
    function deepFreeze(a) {
      Object.keys(a).forEach(c => {if (typeof a[c] == 'object'
&& a[c] !== null && a.hasOwnProperty(c)) deepFreeze(a[c]);});
      return Object.freeze(a);
    }
    const product =
JSON.parse(document.getElementById(“__PRODUCT_DATA__“).textContent)
.products;
    for (var i = product.length - 1; i > 0; i--) {
      var j = Math.floor(Math.random() * (i + 1));
      [product[i], product[j]] = [product[j], product[i]];
    }
    deepFreeze(product);
    product.forEach((c)=>{
      let newTr = document.createElement(“tr“);
      let newTh = document.createElement(“th“);
      newTh.setAttribute(“scope“, “row“);
      newTh.textContent = product.indexOf(c) + 1;
      newTr.appendChild(newTh);
      let newTd = document.createElement(“td“);
      newTd.textContent = c.display_name;
      newTr.appendChild(newTd);
      newTd = document.createElement(“td“);
      newTd.textContent = ‘${c.price}‘;
      newTr.appendChild(newTd);
      newTd = document.createElement(“td“);
      let newInput = document.createElement(“input“);
      newInput.setAttribute(“type“, “number“);
      newInput.setAttribute(“value“, “0“);
      newInput.setAttribute(“min“, “0“);
      newInput.setAttribute(“class“, “form-control“);
      newInput.setAttribute(“id“,
‘product_${product.indexOf(c)}‘);
      newInput.setAttribute(“product“, c.title);
      newTd.appendChild(newInput);
      newTr.appendChild(newTd);
      newTd = document.createElement(“td“);
      newTd.textContent = ‘$0‘;
      newTr.appendChild(newTd);
      newTd = document.createElement(“td“);
      let newButton = document.createElement(“button“);
      newButton.setAttribute(“class“, “btn btn-danger“);
      newButton.textContent = “삭제“;
      newTd.appendChild(newButton);

```

```

        newTr.appendChild(newTd);
        document.getElementById("products").appendChild(newTr);
    })
    let inputs = document.querySelectorAll("input");
    inputs.forEach((c)=>{
        c.addEventListener("change", ()=>{
            let price = product.find((d)=>d.title ===
c.getAttribute("product"));
            var urlParams = new
URLSearchParams(window.location.search);
            urlParams.set(price.title, c.value);
            price = price.price;
            c.parentElement.nextElementSibling.textContent =
`${price * c.value}`;
            let total = 0;
            inputs.forEach((d)=>{
                let price = product.find((e)=>e.title ===
d.getAttribute("product")).price;
                total += price * d.value;
            })
            document.querySelector("h5").textContent = `총액:
${total}`;
            urlParams.delete("total");
            urlParams.set("total", total);
            let newUrl=window.location.href.split('?')[0] + '?' +
urlParams.toString();
            window.history.pushState({ path: newUrl }, '', newUrl);
        })
    })
    let paymentButton =
document.getElementById("paymentButton");
    paymentButton.addEventListener("click", ()=>{
        var urlParams = new
URLSearchParams(window.location.search);
        let total = urlParams.get("total");
        if (total === null) return alert("상품을 선택해주세요.");
        let paymentBody =
document.getElementById("paymentModalBody");
        paymentBody.children[2].children[0].textContent = `#HSPACE
-${Math.floor(Math.random() * 1000000)}`;
        paymentBody.children[3].children[0].textContent =
`${total}`;
        let modal=new
bootstrap.Modal(document.getElementById("paymentModal"));

document.getElementById("paymentConfirm").addEventListener("click
", ()=>{
            modal.hide();
            let modal1=new
bootstrap.Modal(document.getElementById("paymentCompleteModal"))

```

```

paymentCompleteConfirm.addEventListener("click", ()=>{
    modal1.hide();
    window.location.href = "/";
})
fetch('/order', {
    method: 'POST',
    body: JSON.stringify({total:total}),
    headers: {'Content-Type': 'application/json'}
}).then(async response => {
    if (response.ok) thanksalot.innerHTML = await
response.text();
    else thanksalot.innerHTML = "결제에 실패했습니다.";
}).catch(error => {
    thanksalot.innerHTML = "결제할 수 없습니다 다시
시도하세요.";
})
    modal1.show();
})
modal.show();
});
</script>
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>

```

엄청 길다. 일단 html코드에서 의심될만한 부분이 있는지 살펴봤다. 이전에 드림핵 비기너 강의의 워 게임 중, html코드의 시크릿 코드를 통해 플래그를 얻는 문제가 있었는데 이것도 그런 종류일 수도 있다는 생각이 들었다.

음.. 찾아 봤지만 내 눈에는 딱히 의심되는 코드가 보이지 않았다.

5) app.py 파일을 열어보자. Python.File이라 2번 파일과 같은 상황이 벌어질까 무섭지만 일단 실행해보자. 역시 프롬프트에 진입하지 못하고 실행이 종료된다. 하지만 다행이도 이 파일은 메모장으로 열었을 때 정상적으로 출력되었다.



```

from flask import Flask as Django, render_template, request

app = Django(__name__)

@app.route('/')
def index():
    return render_template('cart.html')

@app.route('/order', methods=['POST'])
def order():
    json = request.get_json()
    result = eval(json['total'])
    if not result: return '주문과정 중 에러가 발생했습니다.'
    elif isinstance(result,int): return '주문이 완료되었습니다.'
    return result

if __name__ == '__main__':
    app.run(port=2007)

```

Django? 장고를 말하는 것 같다. 근데 아직 장고라는걸 공부해 보지 않아 이게 대체 무슨 형식의 파일인지 모르겠다. if문에 elif가 쓰이니 C언어인가? 위 아래에서 함수를 호출하고 있는데, 정작 이 코드에서는 호출되는 함수가 정의되어 있지 않다.

6) docker-compose 파일은 YML파일 유형이다. YAML 파일이 yml/ yamll 형식의 확장자를 가진다고 한다. YAML 파일은 인간이 읽을 수 있는 형식으로 데이터를 저장하는 데 사용된다.

```

version: '3'

services:
  web:
    build:
      context: .
      dockerfile: Dockerfile
    ports:
      - "2007:2007"
    volumes:
      - ./app:/app
    environment:
      - FLASK_APP=app.py
      - FLASK_ENV=development

```

7) dockerfile..위의 파일과 이름이 상당히 유사하다. 그런데 2007이라는 숫자가 계속

반복된다. 뭔가 관련이 있긴 하다는 건데..

```
FROM python:3.8

WORKDIR /app
COPY . /app

RUN apt-get update
RUN pip3 install --upgrade pip
RUN pip3 install flask
EXPOSE 2007

ENV FLASK_APP app/app.py

CMD ["flask", "run", "--host", "0.0.0.0", "--port", "2007"]
```

아직 CTF를 풀기에는 기초 지식이 많이 부족한 것 같다. 당장 배운 언어만 떠올려도 C++밖에 떠오르지 않으니.. 일단 드림핵 위게임을 통해, 오늘 열어본 파일들은 어떤 프로그램으로 분석하는지 그리고 어떻게 분석하는지 등을 공부해야 될 것 같다.

지금까지 몇 개의 워 게임을 풀어보긴 했지만, 그것들은 쉬운 난이도라 이렇게 여러 파일들을 분석하며 답을 도출하는 문제는 아니었다. 그래서 오늘 문제는 정말 어떻게 풀어야 할 지 감이 안잡혔던 것 같다.