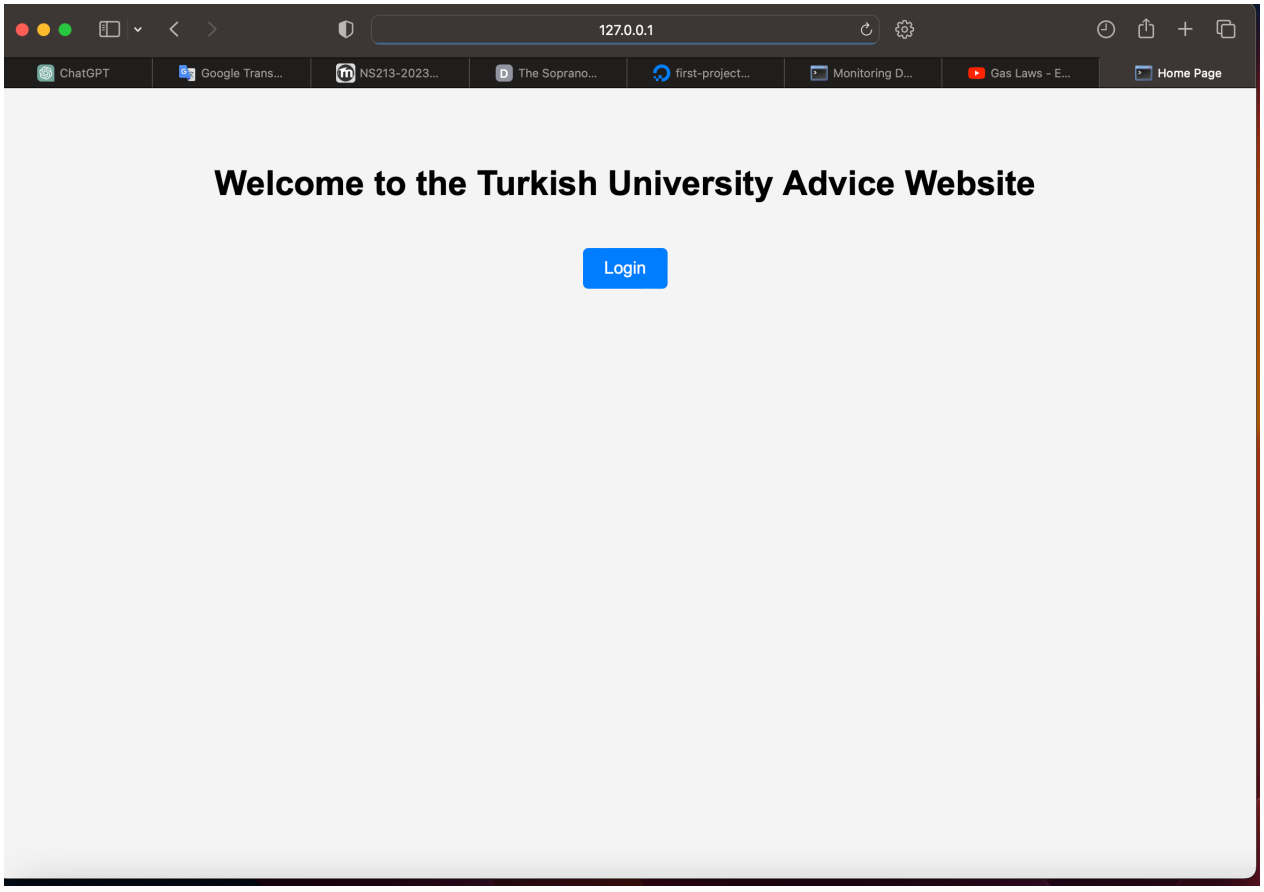
CS437 Assignment 1

Task 6 : Turkish University advice website honeypot

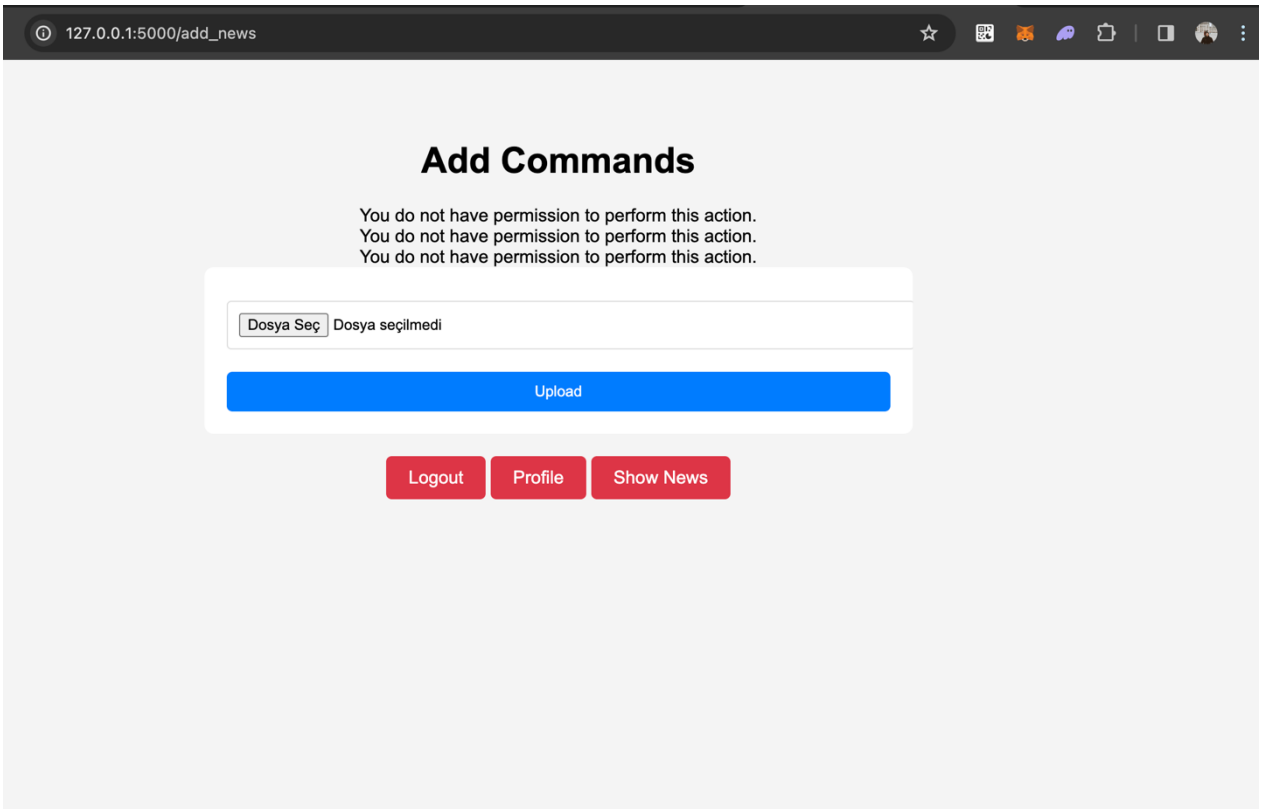
Yağız Gürdamar 22534
Berkant Kartop 28046
Cenk Şire 28075

ScreenShots of Web App:

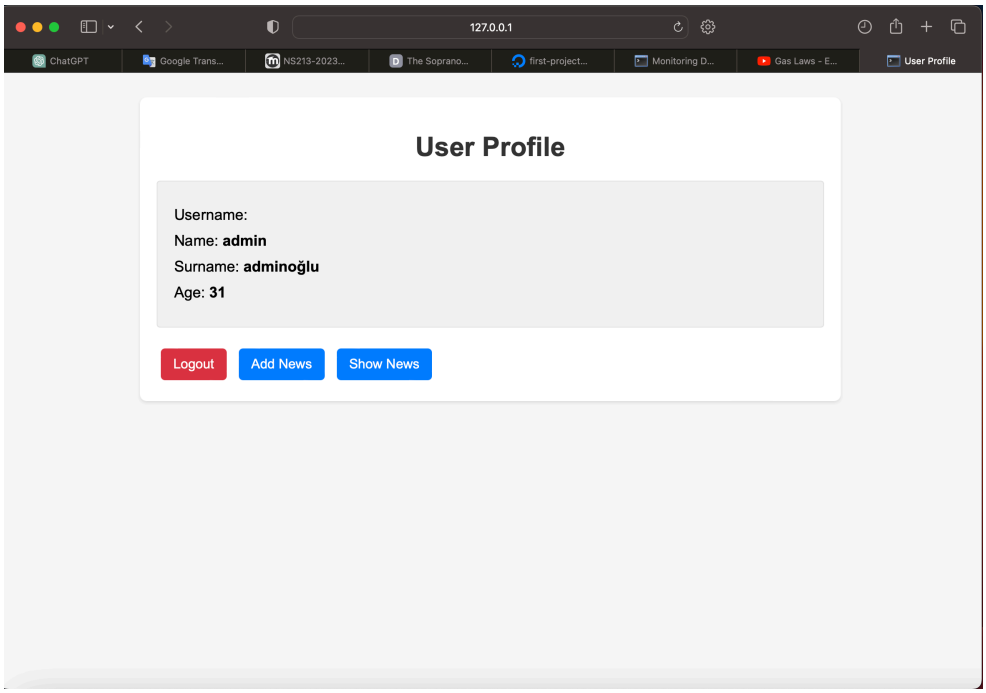
Home:



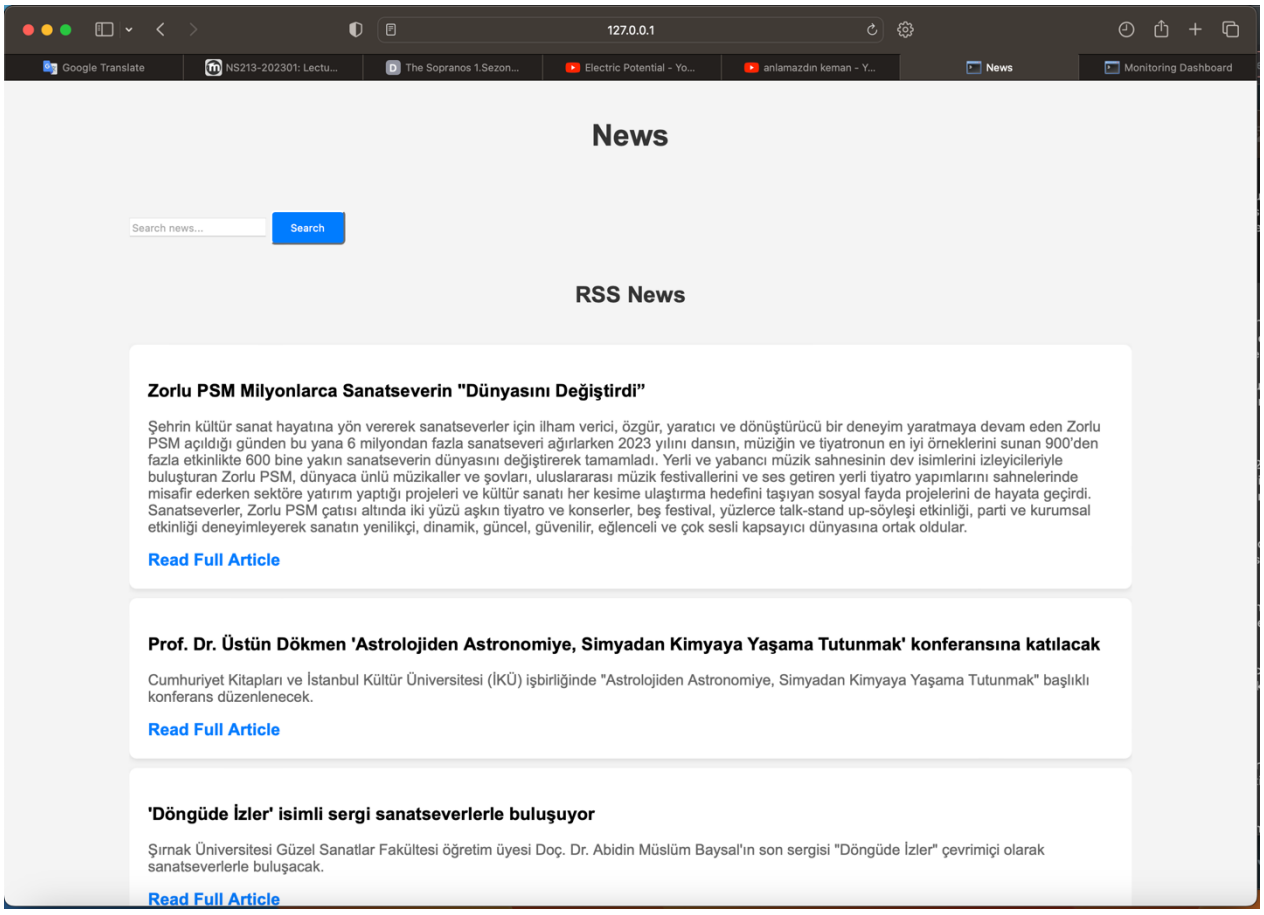
Add Comment Page:



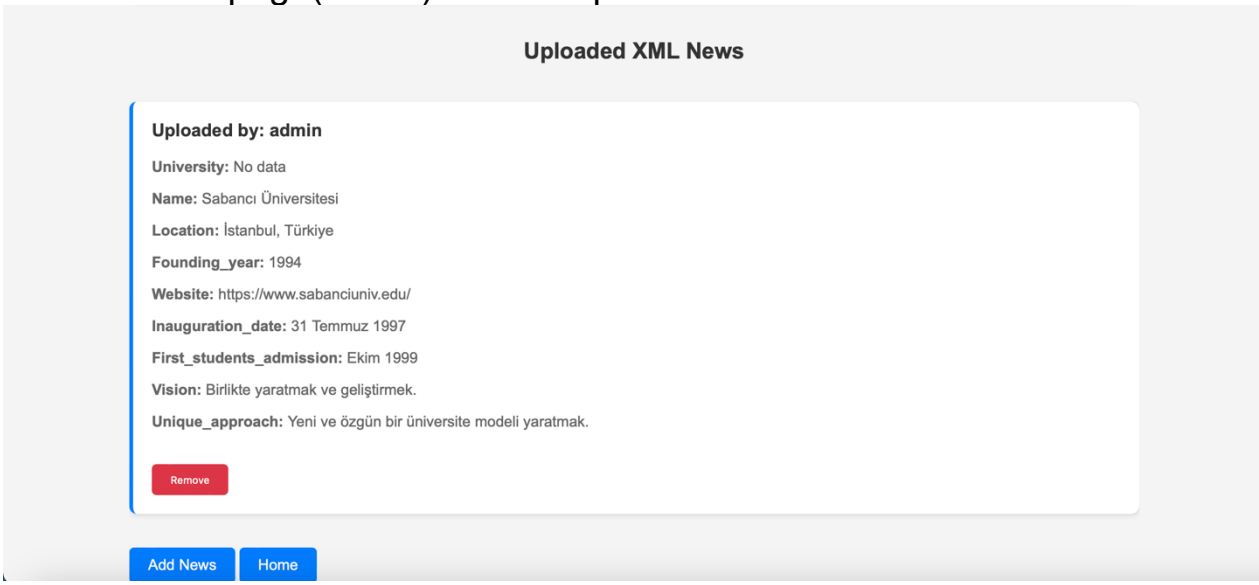
User Profile:



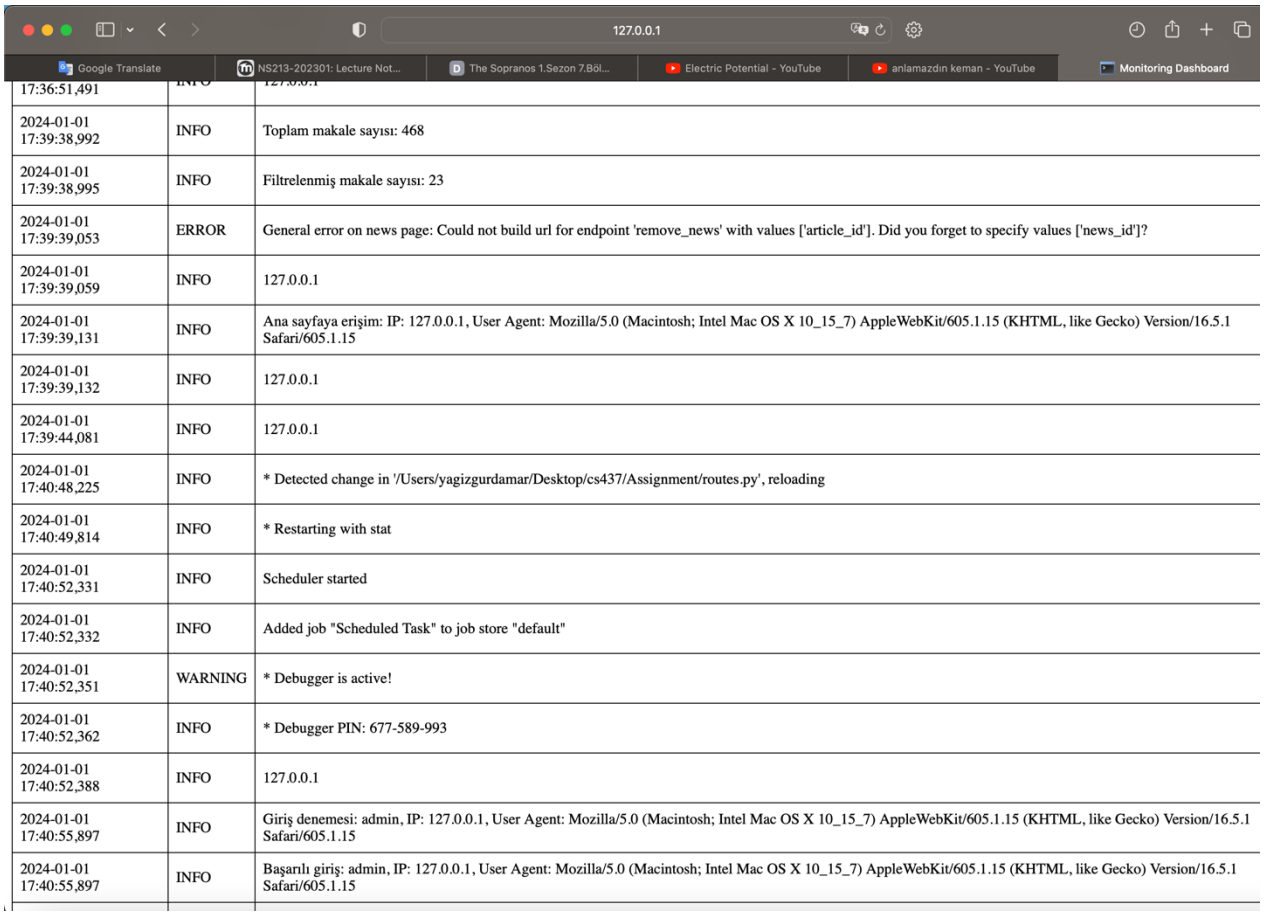
News From RSS Links:



Also in same page(/news) we can uploaded XML content:



Monitoring Dashboard:



17:36:51,491	INFO	127.0.0.1
2024-01-01 17:39:38,992	INFO	Toplam makale sayısı: 468
2024-01-01 17:39:38,995	INFO	Filtrelenmiş makale sayısı: 23
2024-01-01 17:39:39,053	ERROR	General error on news page: Could not build url for endpoint 'remove_news' with values ['article_id']. Did you forget to specify values ['news_id']?
2024-01-01 17:39:39,059	INFO	127.0.0.1
2024-01-01 17:39:39,131	INFO	Ana sayfaya erişim: IP: 127.0.0.1, User Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/16.5.1 Safari/605.1.15
2024-01-01 17:39:39,132	INFO	127.0.0.1
2024-01-01 17:39:44,081	INFO	127.0.0.1
2024-01-01 17:40:48,225	INFO	* Detected change in '/Users/yagizgurdamar/Desktop/cs437/Assignment/routes.py', reloading
2024-01-01 17:40:49,814	INFO	* Restarting with stat
2024-01-01 17:40:52,331	INFO	Scheduler started
2024-01-01 17:40:52,332	INFO	Added job "Scheduled Task" to job store "default"
2024-01-01 17:40:52,351	WARNING	* Debugger is active!
2024-01-01 17:40:52,362	INFO	* Debugger PIN: 677-589-993
2024-01-01 17:40:52,388	INFO	127.0.0.1
2024-01-01 17:40:55,897	INFO	Giriş denemesi: admin, IP: 127.0.0.1, User Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/16.5.1 Safari/605.1.15
2024-01-01 17:40:55,897	INFO	Başarılı giriş: admin, IP: 127.0.0.1, User Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/16.5.1 Safari/605.1.15

Introduction:

This report presents an analysis of the provided web application code, focusing on identifying and explaining various security vulnerabilities and misconfigurations. The application is developed using Python with Flask framework and incorporates functionalities like user registration, login, news feed parsing, and profile management.

Code Overview:

- routes.py: Handles the routing of the application and defines the logic for user authentication, news addition, and profile management.
- models.py: Defines the User and Article models using SQLAlchemy ORM for database interactions.
- func.py: Contains utility functions for RSS feed parsing and user credential verification.
- app.py: Initializes the Flask application, configurations, database, and routes.
- HTML Files: Represent the user interface for various functionalities like adding news, user login, registration, etc.

Identified Vulnerabilities:

1. CWE-315: Cleartext Storage of Sensitive Information in a Cookie
   - Location: routes.py
   - Explanation: The application manages user sessions without explicitly securing the cookie where session information is stored. Flask's default session management mechanism stores session data in a client-side cookie in an unencrypted format, making it susceptible to unauthorized access and tampering.
   - Impact: Attackers can intercept or manipulate the session cookie to impersonate users or steal sensitive information.
   - Mitigation: We can use server-side session management or employ Flask extensions like Flask-Session to store session data securely. Also, use the `secure` and `httponly` flags for cookies to add an extra layer of protection.

2. Failure: XML External Entities (XXE) vulnerability in API
   - Location: routes.py (parse_xml function)
   - Explanation: The application parses XML input using the ElementTree library without disabling external entity processing. This misconfiguration allows an attacker to craft malicious XML data that can include external entities leading to XXE attacks.
   - Impact: XXE attacks can lead to data exfiltration, server-side request forgery (SSRF), or denial of service.
   - Mitigation: We can use a safer XML parser or library configured to disallow external entities. For ElementTree, explicitly disable external entity processing.

```python
def parse_xml(file_path):
    try:
        tree = ET.parse(file_path)
        root = tree.getroot()

        # Tüm elementleri içeren bir listeyi döndür
        news_items = []
        for elem in root.iter():
            if elem.text:  # Eğer element metin içeriyorsa, tag ve text'i list
                text = elem.text.strip()
                news_items.append((elem.tag, text))
            else:  # Eğer element metin içermiyorsa, sadece tag'i ekle
                news_items.append((elem.tag, None))

        return news_items
    except ET.ParseError as e:
        logging.error(f"XML parsing error for file {file_path}: {e}")
        return []
```

3. SQL Injection via User Input
   - Location: routes.py (login and register routes)
   - Explanation: The application constructs SQL queries directly using user-provided input without proper sanitization or parameterization, making it susceptible to SQL injection attacks.
   - Impact: Attackers can execute arbitrary SQL commands leading to unauthorized data access, data manipulation, or even database structure compromise.
   - Mitigation: We can use ORM's parameterized queries feature or employ proper input validation and sanitization techniques.

```python
@app.route('/login', methods=['GET', 'POST'])
def login():
    if request.method == 'POST':
        username = request.form['username']
        password = request.form['password']
        user = User.query.filter_by(username=username).first()
        if user and check_password_hash(user.password, password):
            login_user(user)
            logging.info(f"Successful login: {user.username}")
            return redirect(url_for('add_news'))  # Or wherever you want to r
        else:
            flash('Invalid username or password')
            logging.warning(f"Failed login attempt for {username}")
            return redirect(url_for('login'))
    return render_template('login.html')
```

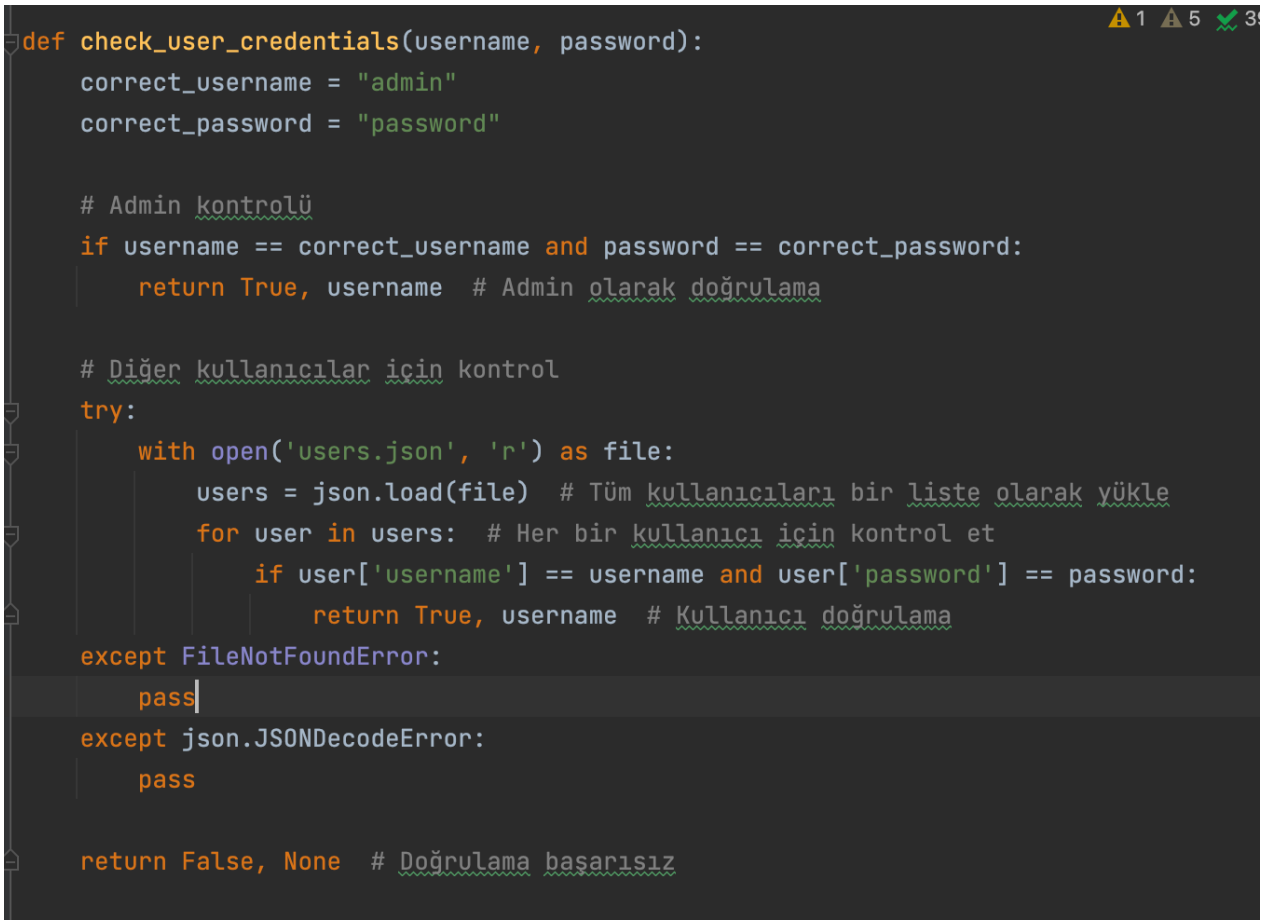
4. Insecure Direct Object References (IDOR)
   - Location: routes.py (remove_news function)
   - Explanation: The application allows users to delete news articles by submitting a direct request with the article ID. There is no proper authorization check to ensure that the user has the right to delete the specific article.
   - Impact: Unauthorized users can manipulate or delete data they shouldn't have access to.
   - Mitigation: We may implement robust access control checks to ensure users can only perform actions on resources they own or have explicit permissions for.

```python
@app.route('/remove_news/<int:article_id>', methods=['POST'])
@login_required
def remove_news(article_id):
    global news_storage
    if current_user.id == 'admin':  # Ensure only admin can delete
        if article_id in news_storage:
            del news_storage[article_id]
            flash('News article removed successfully.', 'success')
        else:
            flash('No article found with the provided ID.', 'error')
    else:
        flash('You do not have permission to perform this action.', 'error')

    return redirect(url_for('show_news'))
```

5. Insufficient Logging and Monitoring
   - Location: Throughout the application
   - Explanation: While the application does some logging, it lacks detailed logging for security-relevant events, and there is no mechanism for real-time alerting or monitoring of suspicious activities.
   - Impact: In the event of an attack, the lack of detailed logs and monitoring can hinder the timely detection and response to security incidents.
   - Mitigation: Implement comprehensive logging of security-relevant events and integrate with a monitoring system that can provide real-time alerts and analysis.

6. Hardcoded Sensitive Information
   - Location: func.py (check_user_credentials function)
   - Explanation: The application contains hardcoded credentials within the codebase. This practice is insecure as it exposes sensitive information in the code.
   - Impact: If the codebase is exposed, attackers can easily find and misuse hardcoded credentials.
   - Mitigation: We might store sensitive information like credentials in environment variables or secure configuration files.



```python
def check_user_credentials(username, password):
    correct_username = "admin"
    correct_password = "password"

    # Admin kontrolü
    if username == correct_username and password == correct_password:
        return True, username  # Admin olarak doğrulama

    # Diğer kullanıcılar için kontrol
    try:
        with open('users.json', 'r') as file:
            users = json.load(file)  # Tüm kullanıcıları bir liste olarak yükle
            for user in users:  # Her bir kullanıcı için kontrol et
                if user['username'] == username and user['password'] == password:
                    return True, username  # Kullanıcı doğrulama
    except FileNotFoundError:
        pass
    except json.JSONDecodeError:
        pass

    return False, None  # Doğrulama başarısız
```

How to Demonstrate Vulnerabilities:

Conclusion:

The identified vulnerabilities and misconfigurations represent significant security risks. It's crucial to address these issues by implementing the suggested mitigation strategies. Additionally, adopting a secure coding practice and regular security audits can further enhance the application's security posture.

Testing:

Flake8:

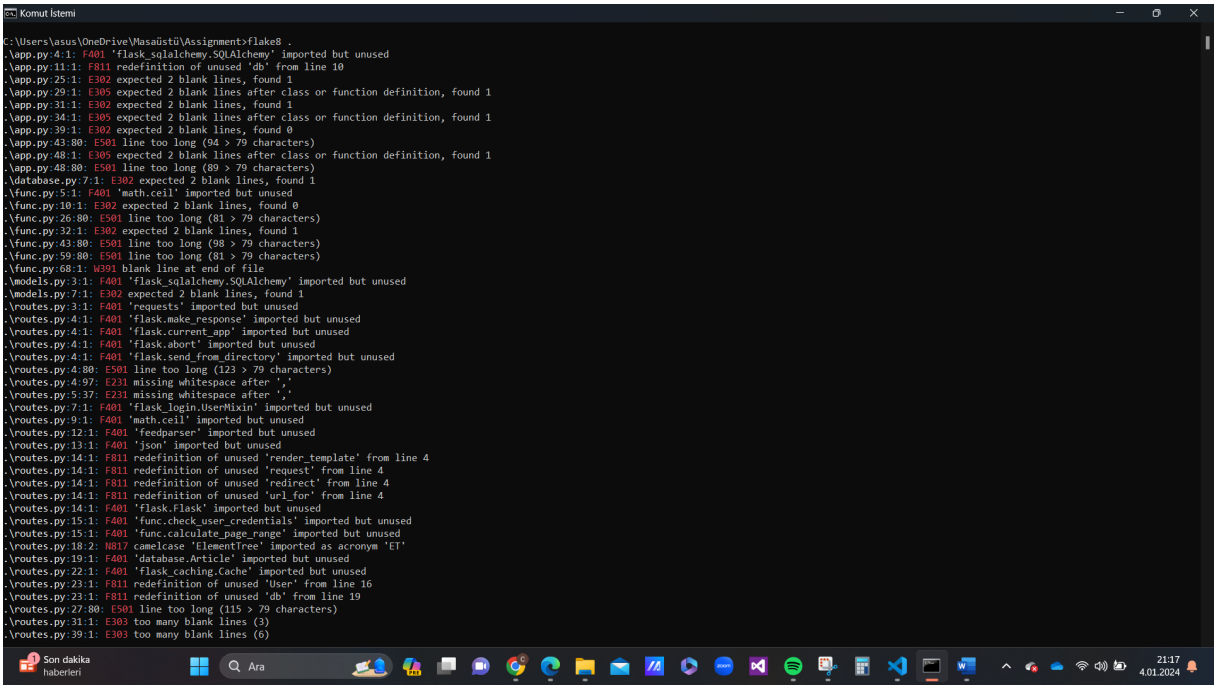
- **OS:** Windows
- **Findings:**

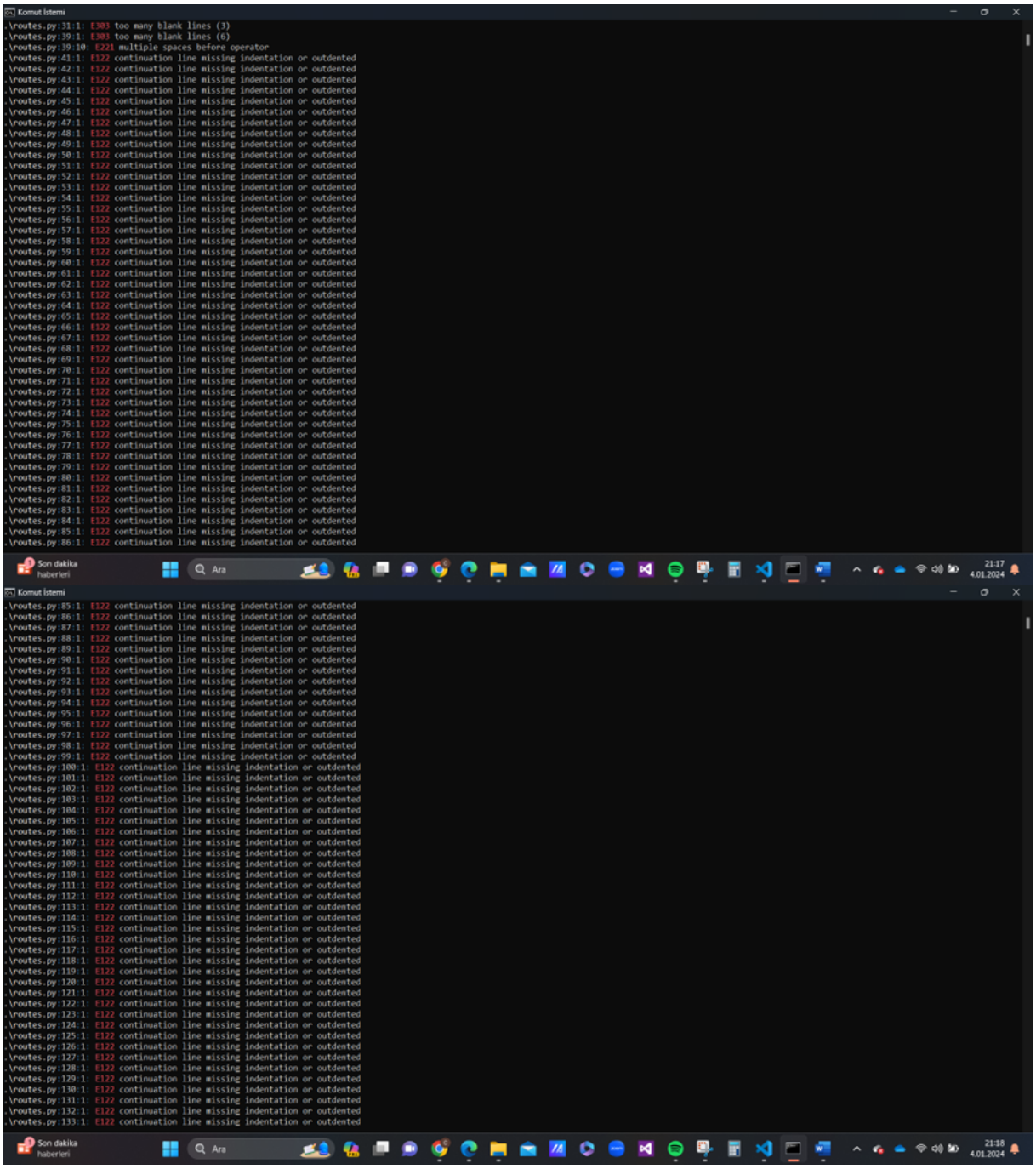
  ○ F401: Unused Imports - Multiple instances of modules or objects imported but not used, leading to namespace bloat.
  ○ F811: Redefinition of Unused Variables - Variables redefined without being used, indicating poor code maintenance.
  ○ E302 & E305: Spacing Issues - Incorrect number of blank lines around functions or class definitions, violating PEP 8 style guidelines.
  ○ E501: Line Length - Lines of code exceeding the recommended length of 79 characters, affecting readability and maintenance.
  ○ E231: Whitespace Issues - Missing whitespace after commas, a style issue.
  ○ E122 & E131: Indentation Issues - Improper indentation for continuation lines, impacting code readability.
  ○ W391: Blank Line at End of File - Unnecessary blank line at the file's end.

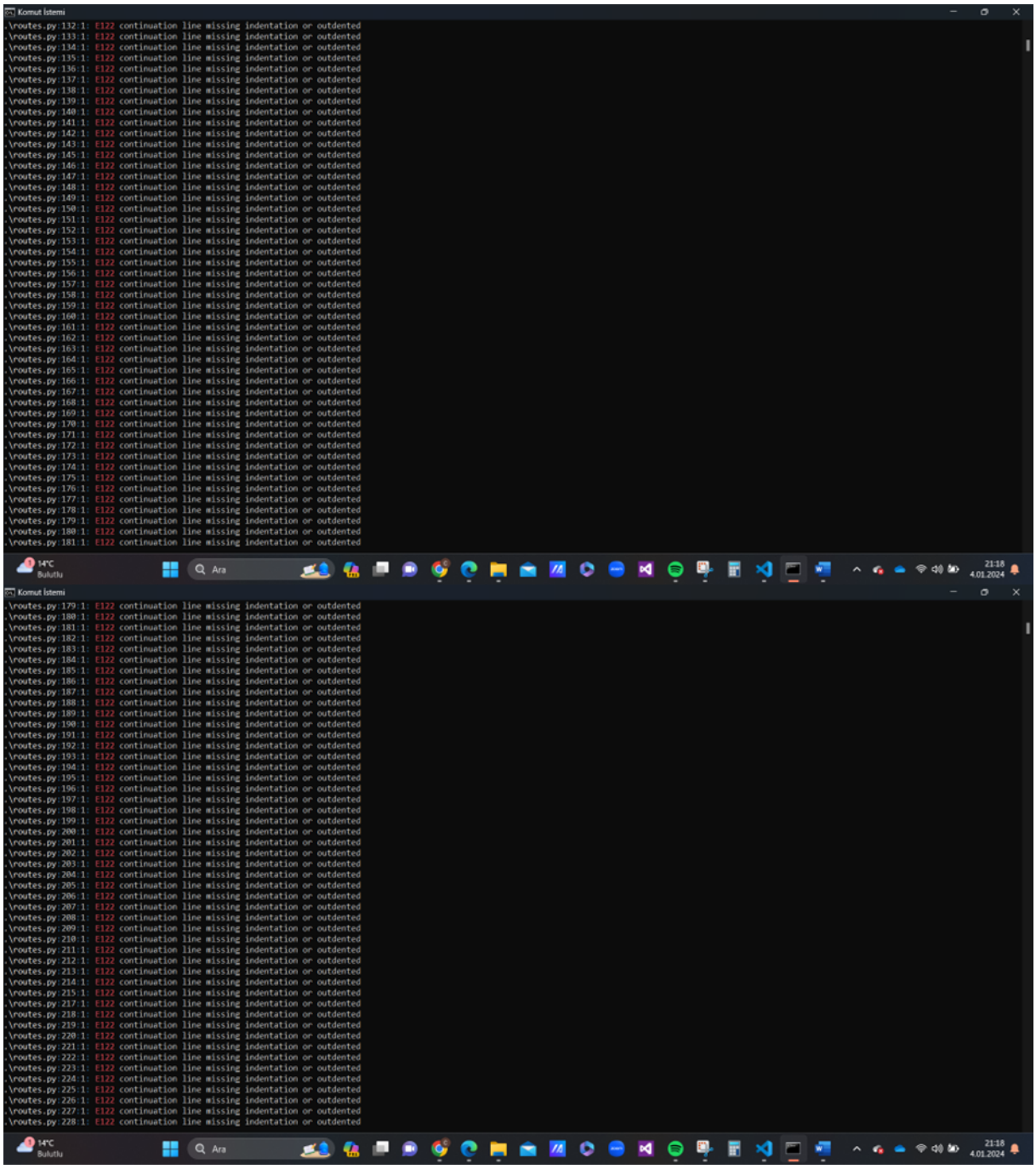
- **Conclusion:**

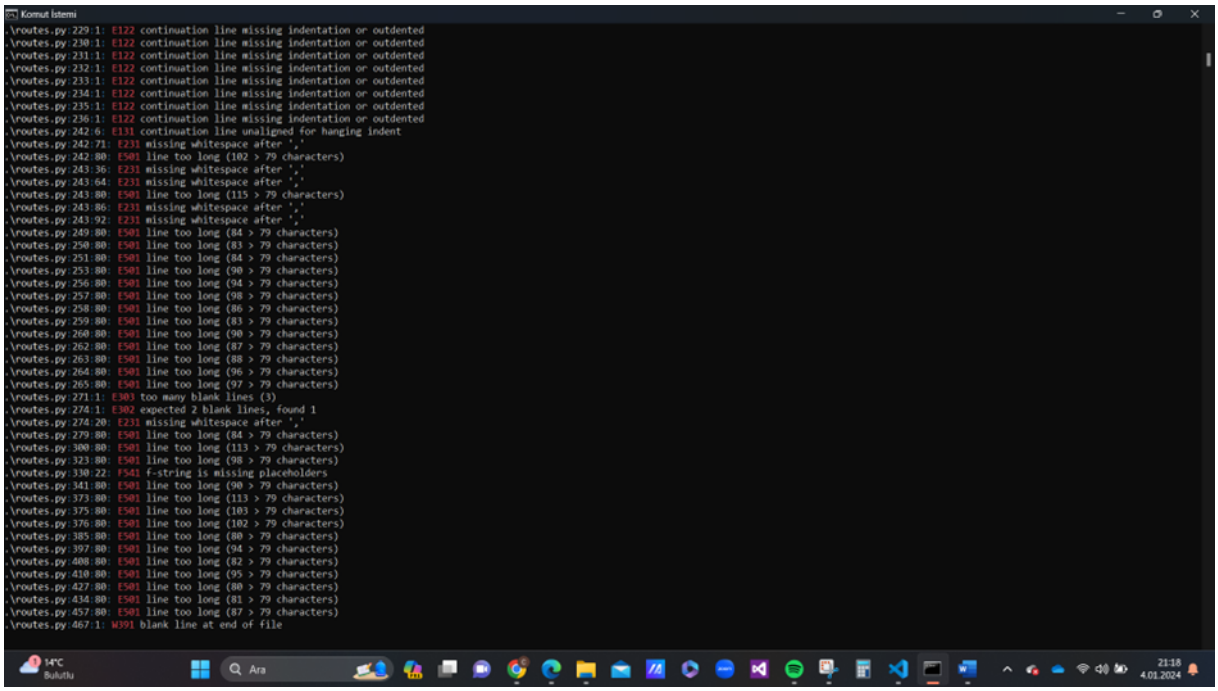
The flake8 tool primarily focuses on code quality and adherence to PEP 8 style guidelines, rather than directly identifying security vulnerabilities. While these issues, such as unused imports, redefinition of variables, spacing, line length, whitespace, and indentation issues, do not represent direct security risks, they can indirectly impact the overall security posture by affecting code clarity and maintainability.

- **Screenshots:**




Komut İstemi
.\routes.py:31:1: E303 too many blank lines (3)
.\routes.py:39:1: E303 too many blank lines (6)
.\routes.py:39:10: E221 multiple spaces before operator
.\routes.py:41:1: E122 continuation line missing indentation or outdented
.\routes.py:42:1: E122 continuation line missing indentation or outdented
.\routes.py:43:1: E122 continuation line missing indentation or outdented
.\routes.py:44:1: E122 continuation line missing indentation or outdented
.\routes.py:45:1: E122 continuation line missing indentation or outdented
.\routes.py:46:1: E122 continuation line missing indentation or outdented
.\routes.py:47:1: E122 continuation line missing indentation or outdented
.\routes.py:48:1: E122 continuation line missing indentation or outdented
.\routes.py:49:1: E122 continuation line missing indentation or outdented
.\routes.py:50:1: E122 continuation line missing indentation or outdented
.\routes.py:51:1: E122 continuation line missing indentation or outdented
.\routes.py:52:1: E122 continuation line missing indentation or outdented
.\routes.py:53:1: E122 continuation line missing indentation or outdented
.\routes.py:54:1: E122 continuation line missing indentation or outdented
.\routes.py:55:1: E122 continuation line missing indentation or outdented
.\routes.py:56:1: E122 continuation line missing indentation or outdented
.\routes.py:57:1: E122 continuation line missing indentation or outdented
.\routes.py:58:1: E122 continuation line missing indentation or outdented
.\routes.py:59:1: E122 continuation line missing indentation or outdented
.\routes.py:60:1: E122 continuation line missing indentation or outdented
.\routes.py:61:1: E122 continuation line missing indentation or outdented
.\routes.py:62:1: E122 continuation line missing indentation or outdented
.\routes.py:63:1: E122 continuation line missing indentation or outdented
.\routes.py:64:1: E122 continuation line missing indentation or outdented
.\routes.py:65:1: E122 continuation line missing indentation or outdented
.\routes.py:66:1: E122 continuation line missing indentation or outdented
.\routes.py:67:1: E122 continuation line missing indentation or outdented
.\routes.py:68:1: E122 continuation line missing indentation or outdented
.\routes.py:69:1: E122 continuation line missing indentation or outdented
.\routes.py:70:1: E122 continuation line missing indentation or outdented
.\routes.py:71:1: E122 continuation line missing indentation or outdented
.\routes.py:72:1: E122 continuation line missing indentation or outdented
.\routes.py:73:1: E122 continuation line missing indentation or outdented
.\routes.py:74:1: E122 continuation line missing indentation or outdented
.\routes.py:75:1: E122 continuation line missing indentation or outdented
.\routes.py:76:1: E122 continuation line missing indentation or outdented
.\routes.py:77:1: E122 continuation line missing indentation or outdented
.\routes.py:78:1: E122 continuation line missing indentation or outdented
.\routes.py:79:1: E122 continuation line missing indentation or outdented
.\routes.py:80:1: E122 continuation line missing indentation or outdented
.\routes.py:81:1: E122 continuation line missing indentation or outdented
.\routes.py:82:1: E122 continuation line missing indentation or outdented
.\routes.py:83:1: E122 continuation line missing indentation or outdented
.\routes.py:84:1: E122 continuation line missing indentation or outdented
.\routes.py:85:1: E122 continuation line missing indentation or outdented
.\routes.py:86:1: E122 continuation line missing indentation or outdented
Son dakika haberleri
Ara
21:17
4.01.2024
Komut İstemi
.\routes.py:85:1: E122 continuation line missing indentation or outdented
.\routes.py:86:1: E122 continuation line missing indentation or outdented
.\routes.py:87:1: E122 continuation line missing indentation or outdented
.\routes.py:88:1: E122 continuation line missing indentation or outdented
.\routes.py:89:1: E122 continuation line missing indentation or outdented
.\routes.py:90:1: E122 continuation line missing indentation or outdented
.\routes.py:91:1: E122 continuation line missing indentation or outdented
.\routes.py:92:1: E122 continuation line missing indentation or outdented
.\routes.py:93:1: E122 continuation line missing indentation or outdented
.\routes.py:94:1: E122 continuation line missing indentation or outdented
.\routes.py:95:1: E122 continuation line missing indentation or outdented
.\routes.py:96:1: E122 continuation line missing indentation or outdented
.\routes.py:97:1: E122 continuation line missing indentation or outdented
.\routes.py:98:1: E122 continuation line missing indentation or outdented
.\routes.py:99:1: E122 continuation line missing indentation or outdented
.\routes.py:100:1: E122 continuation line missing indentation or outdented
.\routes.py:101:1: E122 continuation line missing indentation or outdented
.\routes.py:102:1: E122 continuation line missing indentation or outdented
.\routes.py:103:1: E122 continuation line missing indentation or outdented
.\routes.py:104:1: E122 continuation line missing indentation or outdented
.\routes.py:105:1: E122 continuation line missing indentation or outdented
.\routes.py:106:1: E122 continuation line missing indentation or outdented
.\routes.py:107:1: E122 continuation line missing indentation or outdented
.\routes.py:108:1: E122 continuation line missing indentation or outdented
.\routes.py:109:1: E122 continuation line missing indentation or outdented
.\routes.py:110:1: E122 continuation line missing indentation or outdented
.\routes.py:111:1: E122 continuation line missing indentation or outdented
.\routes.py:112:1: E122 continuation line missing indentation or outdented
.\routes.py:113:1: E122 continuation line missing indentation or outdented
.\routes.py:114:1: E122 continuation line missing indentation or outdented
.\routes.py:115:1: E122 continuation line missing indentation or outdented
.\routes.py:116:1: E122 continuation line missing indentation or outdented
.\routes.py:117:1: E122 continuation line missing indentation or outdented
.\routes.py:118:1: E122 continuation line missing indentation or outdented
.\routes.py:119:1: E122 continuation line missing indentation or outdented
.\routes.py:120:1: E122 continuation line missing indentation or outdented
.\routes.py:121:1: E122 continuation line missing indentation or outdented
.\routes.py:122:1: E122 continuation line missing indentation or outdented
.\routes.py:123:1: E122 continuation line missing indentation or outdented
.\routes.py:124:1: E122 continuation line missing indentation or outdented
.\routes.py:125:1: E122 continuation line missing indentation or outdented
.\routes.py:126:1: E122 continuation line missing indentation or outdented
.\routes.py:127:1: E122 continuation line missing indentation or outdented
.\routes.py:128:1: E122 continuation line missing indentation or outdented
.\routes.py:129:1: E122 continuation line missing indentation or outdented
.\routes.py:130:1: E122 continuation line missing indentation or outdented
.\routes.py:131:1: E122 continuation line missing indentation or outdented
.\routes.py:132:1: E122 continuation line missing indentation or outdented
.\routes.py:133:1: E122 continuation line missing indentation or outdented
Son dakika haberleri
Ara
21:18
4.01.2024


Komut İstemi
.\routes.py:132:1: E122 continuation line missing indentation or outdented
.\routes.py:133:1: E122 continuation line missing indentation or outdented
.\routes.py:134:1: E122 continuation line missing indentation or outdented
.\routes.py:135:1: E122 continuation line missing indentation or outdented
.\routes.py:136:1: E122 continuation line missing indentation or outdented
.\routes.py:137:1: E122 continuation line missing indentation or outdented
.\routes.py:138:1: E122 continuation line missing indentation or outdented
.\routes.py:139:1: E122 continuation line missing indentation or outdented
.\routes.py:140:1: E122 continuation line missing indentation or outdented
.\routes.py:141:1: E122 continuation line missing indentation or outdented
.\routes.py:142:1: E122 continuation line missing indentation or outdented
.\routes.py:143:1: E122 continuation line missing indentation or outdented
.\routes.py:145:1: E122 continuation line missing indentation or outdented
.\routes.py:146:1: E122 continuation line missing indentation or outdented
.\routes.py:147:1: E122 continuation line missing indentation or outdented
.\routes.py:148:1: E122 continuation line missing indentation or outdented
.\routes.py:149:1: E122 continuation line missing indentation or outdented
.\routes.py:150:1: E122 continuation line missing indentation or outdented
.\routes.py:151:1: E122 continuation line missing indentation or outdented
.\routes.py:152:1: E122 continuation line missing indentation or outdented
.\routes.py:153:1: E122 continuation line missing indentation or outdented
.\routes.py:154:1: E122 continuation line missing indentation or outdented
.\routes.py:155:1: E122 continuation line missing indentation or outdented
.\routes.py:156:1: E122 continuation line missing indentation or outdented
.\routes.py:157:1: E122 continuation line missing indentation or outdented
.\routes.py:158:1: E122 continuation line missing indentation or outdented
.\routes.py:159:1: E122 continuation line missing indentation or outdented
.\routes.py:160:1: E122 continuation line missing indentation or outdented
.\routes.py:161:1: E122 continuation line missing indentation or outdented
.\routes.py:162:1: E122 continuation line missing indentation or outdented
.\routes.py:163:1: E122 continuation line missing indentation or outdented
.\routes.py:164:1: E122 continuation line missing indentation or outdented
.\routes.py:165:1: E122 continuation line missing indentation or outdented
.\routes.py:166:1: E122 continuation line missing indentation or outdented
.\routes.py:167:1: E122 continuation line missing indentation or outdented
.\routes.py:168:1: E122 continuation line missing indentation or outdented
.\routes.py:169:1: E122 continuation line missing indentation or outdented
.\routes.py:170:1: E122 continuation line missing indentation or outdented
.\routes.py:171:1: E122 continuation line missing indentation or outdented
.\routes.py:172:1: E122 continuation line missing indentation or outdented
.\routes.py:173:1: E122 continuation line missing indentation or outdented
.\routes.py:174:1: E122 continuation line missing indentation or outdented
.\routes.py:175:1: E122 continuation line missing indentation or outdented
.\routes.py:176:1: E122 continuation line missing indentation or outdented
.\routes.py:177:1: E122 continuation line missing indentation or outdented
.\routes.py:178:1: E122 continuation line missing indentation or outdented
.\routes.py:179:1: E122 continuation line missing indentation or outdented
.\routes.py:180:1: E122 continuation line missing indentation or outdented
.\routes.py:181:1: E122 continuation line missing indentation or outdented
14°C
Bulutlu
Ara
21:18
4.01.2024
Komut İstemi
.\routes.py:179:1: E122 continuation line missing indentation or outdented
.\routes.py:180:1: E122 continuation line missing indentation or outdented
.\routes.py:181:1: E122 continuation line missing indentation or outdented
.\routes.py:182:1: E122 continuation line missing indentation or outdented
.\routes.py:183:1: E122 continuation line missing indentation or outdented
.\routes.py:184:1: E122 continuation line missing indentation or outdented
.\routes.py:185:1: E122 continuation line missing indentation or outdented
.\routes.py:186:1: E122 continuation line missing indentation or outdented
.\routes.py:187:1: E122 continuation line missing indentation or outdented
.\routes.py:188:1: E122 continuation line missing indentation or outdented
.\routes.py:189:1: E122 continuation line missing indentation or outdented
.\routes.py:190:1: E122 continuation line missing indentation or outdented
.\routes.py:191:1: E122 continuation line missing indentation or outdented
.\routes.py:192:1: E122 continuation line missing indentation or outdented
.\routes.py:193:1: E122 continuation line missing indentation or outdented
.\routes.py:194:1: E122 continuation line missing indentation or outdented
.\routes.py:195:1: E122 continuation line missing indentation or outdented
.\routes.py:196:1: E122 continuation line missing indentation or outdented
.\routes.py:197:1: E122 continuation line missing indentation or outdented
.\routes.py:198:1: E122 continuation line missing indentation or outdented
.\routes.py:199:1: E122 continuation line missing indentation or outdented
.\routes.py:200:1: E122 continuation line missing indentation or outdented
.\routes.py:201:1: E122 continuation line missing indentation or outdented
.\routes.py:202:1: E122 continuation line missing indentation or outdented
.\routes.py:203:1: E122 continuation line missing indentation or outdented
.\routes.py:204:1: E122 continuation line missing indentation or outdented
.\routes.py:205:1: E122 continuation line missing indentation or outdented
.\routes.py:206:1: E122 continuation line missing indentation or outdented
.\routes.py:207:1: E122 continuation line missing indentation or outdented
.\routes.py:208:1: E122 continuation line missing indentation or outdented
.\routes.py:209:1: E122 continuation line missing indentation or outdented
.\routes.py:210:1: E122 continuation line missing indentation or outdented
.\routes.py:211:1: E122 continuation line missing indentation or outdented
.\routes.py:212:1: E122 continuation line missing indentation or outdented
.\routes.py:213:1: E122 continuation line missing indentation or outdented
.\routes.py:214:1: E122 continuation line missing indentation or outdented
.\routes.py:215:1: E122 continuation line missing indentation or outdented
.\routes.py:217:1: E122 continuation line missing indentation or outdented
.\routes.py:218:1: E122 continuation line missing indentation or outdented
.\routes.py:219:1: E122 continuation line missing indentation or outdented
.\routes.py:220:1: E122 continuation line missing indentation or outdented
.\routes.py:221:1: E122 continuation line missing indentation or outdented
.\routes.py:222:1: E122 continuation line missing indentation or outdented
.\routes.py:223:1: E122 continuation line missing indentation or outdented
.\routes.py:224:1: E122 continuation line missing indentation or outdented
.\routes.py:225:1: E122 continuation line missing indentation or outdented
.\routes.py:226:1: E122 continuation line missing indentation or outdented
.\routes.py:227:1: E122 continuation line missing indentation or outdented
.\routes.py:228:1: E122 continuation line missing indentation or outdented
14°C
Bulutlu
Ara
21:18
4.01.2024



Bandit:

- **OS:** Windows
- **Findings:**
    - B105: Hardcoded Password String
    - B201: Flask Debug True
    - B410 & B405: Using etree and ElementTree to Parse Untrusted XML Data
    - B314: Using xml.etree.ElementTree.parse

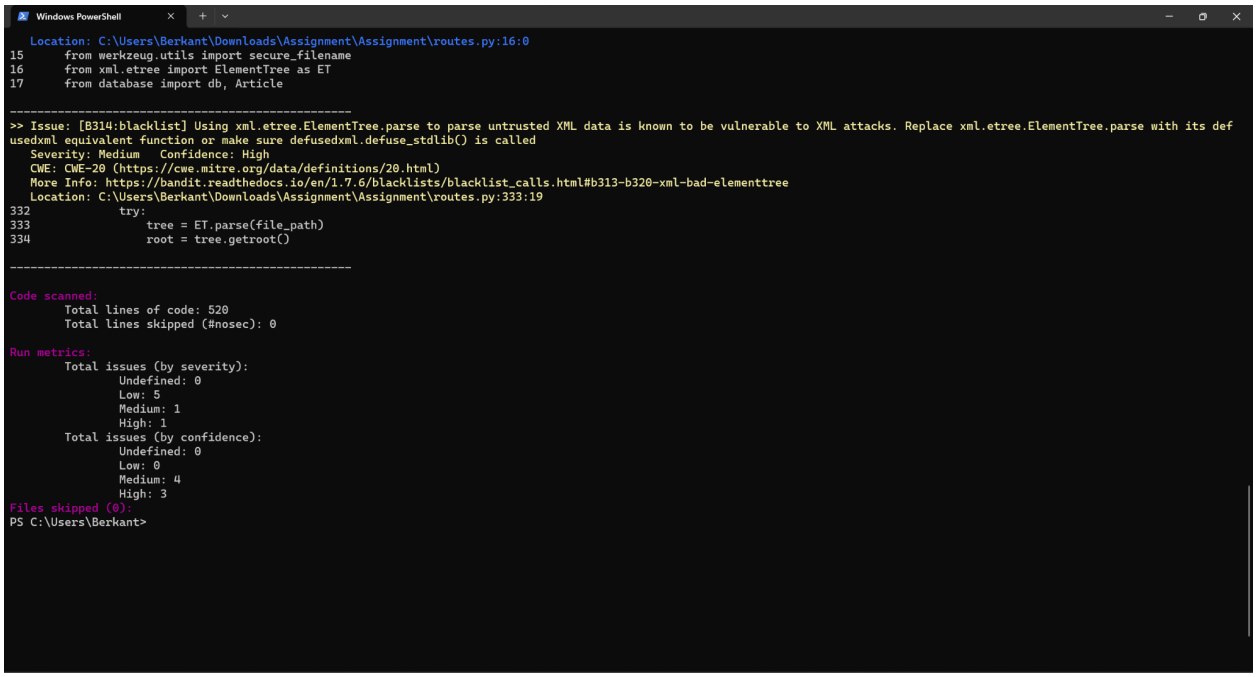
- **Conclusion:**

    It can be said that the bandit is capable of finding vulnerabilities in a python code partially successfully, but not completely since the target vulnerabilities of the chosen task were, CWE-315 Cleartext Storage of Sensitive Information in a Cookie & XML External Entities (XXE) vulnerability in API Failure. It is capable of detecting parsation of an untrusted XML data, However, it does not seem to have detected the cleartext storage of sensitive information in a cookie (CWE-315).

    Bandit was really easy to install and use. Documentation wirtten by the owners was clear. Results were listed separately which makes them easier to understand.
- **Screenshots:**

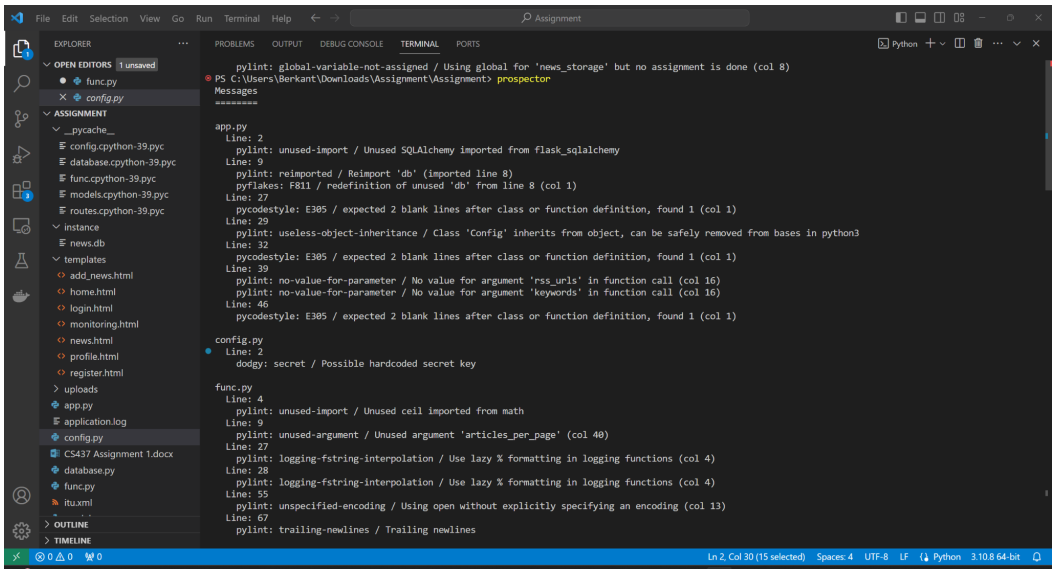

Windows PowerShell
Installing collected packages: tomli
Successfully installed tomli-2.0.1
[notice] A new release of pip available: 22.2.2 -> 23.3.2
[notice] To update, run: python.exe -m pip install --upgrade pip
PS C:\Users\Berkant> bandit -r C:\Users\Berkant\Downloads\Assignment\Assignment
[main] INFO profile include tests: None
[main] INFO profile exclude tests: None
[main] INFO cli include tests: None
[main] INFO cli exclude tests: None
[main] INFO running on Python 3.10.8
Run started:2024-01-02 20:06:00.438686
Test results:
>> Issue: [B105:hardcoded_password_string] Possible hardcoded password: 'your_secret_key'
Severity: Low Confidence: Medium
CWE: CWE-259 (https://cwe.mitre.org/data/definitions/259.html)
More Info: https://bandit.readthedocs.io/en/1.7.6/plugins/b105_hardcoded_password_string.html
Location: C:\Users\Berkant\Downloads\Assignment\Assignment\app.py:12:17
11 app = Flask(__name__)
12 app.secret_key = 'your_secret_key'
13
>> Issue: [B201:flask_debug_true] A Flask app appears to be run with debug=True, which exposes the Werkzeug debugger and allows the execution of arbitrary code.
Severity: High Confidence: Medium
CWE: CWE-94 (https://cwe.mitre.org/data/definitions/94.html)
More Info: https://bandit.readthedocs.io/en/1.7.6/plugins/b201_flask_debug_true.html
Location: C:\Users\Berkant\Downloads\Assignment\Assignment\app.py:54:4
53 db.create_all() # Create tables based on models
54 app.run(debug=True)
>> Issue: [B105:hardcoded_password_string] Possible hardcoded password: 'your_secret_key'
Severity: Low Confidence: Medium
CWE: CWE-259 (https://cwe.mitre.org/data/definitions/259.html)
More Info: https://bandit.readthedocs.io/en/1.7.6/plugins/b105_hardcoded_password_string.html
Location: C:\Users\Berkant\Downloads\Assignment\Assignment\config.py:2:13
1 # Example: config.py
2 SECRET_KEY = 'your_secret_key'
3 # More configuration options can be added here
>> Issue: [B105:hardcoded_password_string] Possible hardcoded password: 'password'

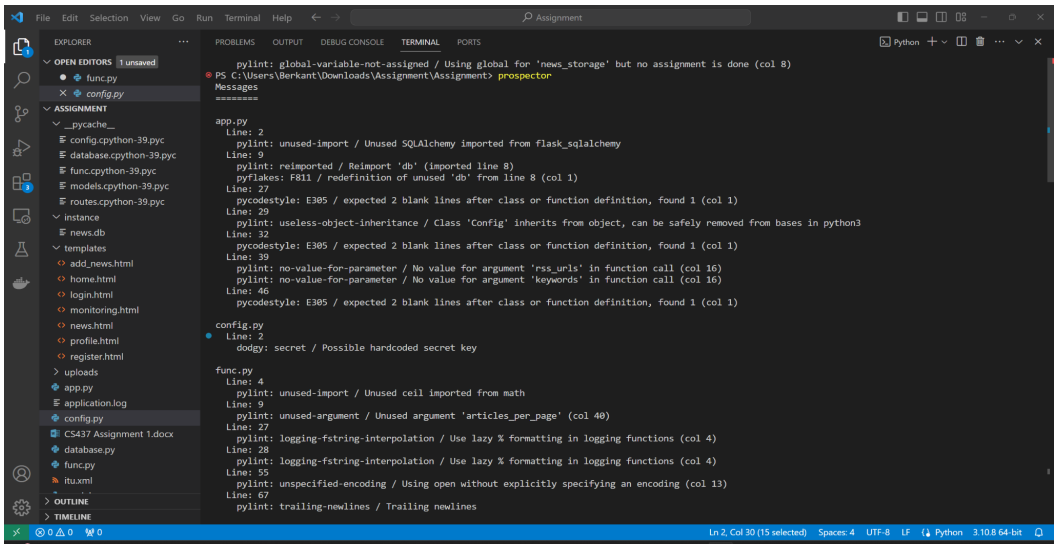
Windows PowerShell
>> Issue: [B105:hardcoded_password_string] Possible hardcoded password: 'password'
Severity: Low Confidence: Medium
CWE: CWE-259 (https://cwe.mitre.org/data/definitions/259.html)
More Info: https://bandit.readthedocs.io/en/1.7.6/plugins/b105_hardcoded_password_string.html
Location: C:\Users\Berkant\Downloads\Assignment\Assignment\func.py:47:23
46 correct_username = "admin"
47 correct_password = "password"
48
>> Issue: [B410:blacklist] Using etree to parse untrusted XML data is known to be vulnerable to XML attacks. Replace etree with the equivalent defusedxml package.
Severity: Low Confidence: High
CWE: CWE-20 (https://cwe.mitre.org/data/definitions/20.html)
More Info: https://bandit.readthedocs.io/en/1.7.6/blacklists/blacklist_imports.html#b410-import-lxml
Location: C:\Users\Berkant\Downloads\Assignment\Assignment\routes.py:6:0
5 from flask_login import UserMixin
6 from lxml import etree
7 from math import ceil
>> Issue: [B405:blacklist] Using ElementTree to parse untrusted XML data is known to be vulnerable to XML attacks. Replace ElementTree with the equivalent defusedxml package, or make sure defusedxml.defuse_stdlib() is called.
Severity: Low Confidence: High
CWE: CWE-20 (https://cwe.mitre.org/data/definitions/20.html)
More Info: https://bandit.readthedocs.io/en/1.7.6/blacklists/blacklist_imports.html#b405-import-xml-etree
Location: C:\Users\Berkant\Downloads\Assignment\Assignment\routes.py:16:0
15 from werkzeug.utils import secure_filename
16 from xml.etree import ElementTree as ET
17 from database import db, Article
>> Issue: [B314:blacklist] Using xml.etree.ElementTree.parse to parse untrusted XML data is known to be vulnerable to XML attacks. Replace xml.etree.ElementTree.parse with its defusedxml equivalent function or make sure defusedxml.defuse_stdlib() is called
Severity: Medium Confidence: High
CWE: CWE-20 (https://cwe.mitre.org/data/definitions/20.html)
More Info: https://bandit.readthedocs.io/en/1.7.6/blacklists/blacklist_calls.html#b313-b320-xml-bad-elementtree
Location: C:\Users\Berkant\Downloads\Assignment\Assignment\routes.py:333:19
332 try:
333 tree = ET.parse(file_path)
334 root = tree.getroot()

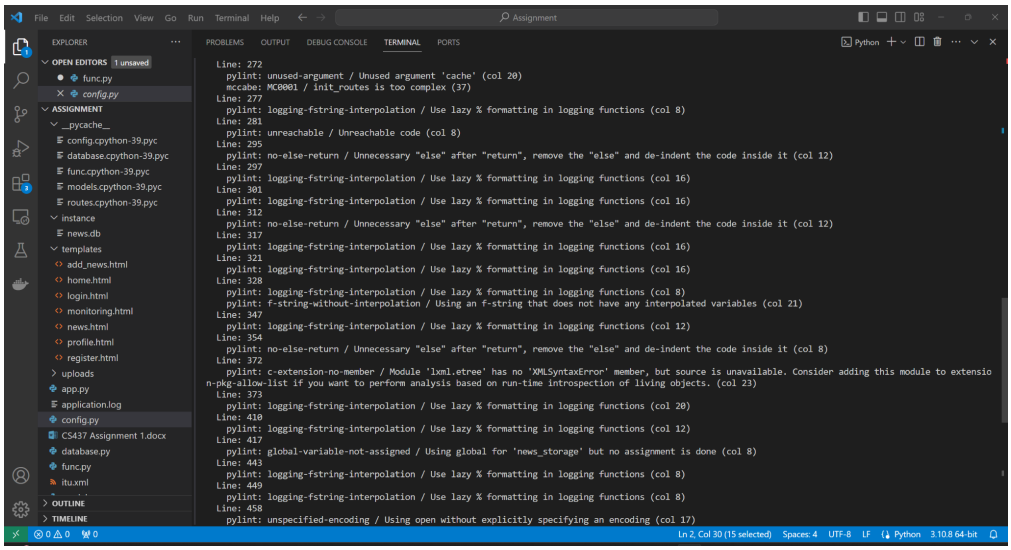


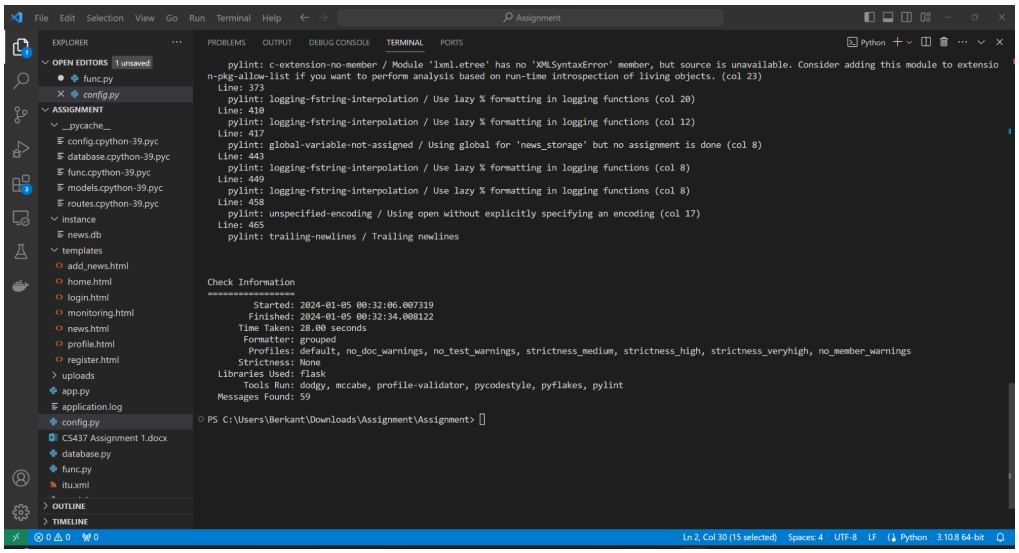
Prospector:

- **OS:** Windows
- **Findings:**
  - Beside the “Hardcoded Secret Key Warning” most of the outcome that prospector give is, factors that related to complexity or efficiency of the code such as, unused imports, re-imported modules, etc.
- **Conclusion:**
  - The finding that mentions “Hardcoded Secret Key Warning” is related to CWE-315 due to usage of the secret key for cookie encryption, but still it is not offering any findings on XXE.
  - Even if the prospector can be beneficial for improving a project make it more secure and efficient, it would be better to use different analyser in order to detect CWE-315 and XXE vulnurabilities.
  - Installation and usage process of the Prospector was easy. The main problem was results were listed as line by line rather than catagorized as topics, so repeated sentences and finding can occur.


File Edit Selection View Go Run Terminal Help
Assignment
EXPLORER
OPEN EDITORS 1 unsaved
func.py
config.py
ASSIGNMENT
__pycache__
config.cpython-39.pyc
database.cpython-39.pyc
func.cpython-39.pyc
models.cpython-39.pyc
routes.cpython-39.pyc
instance
news.db
templates
add_news.html
home.html
login.html
monitoring.html
news.html
profile.html
register.html
uploads
app.py
application.log
config.py
CS437 Assignment 1.docx
database.py
func.py
itu.xml
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Python
pylint: global-variable-not-assigned / Using global for 'news_storage' but no assignment is done (col 8)
PS C:\Users\Berkant\Downloads\Assignment\Assignment> prospector
Messages
========
app.py
Line: 2
pylint: unused-import / Unused SQLAlchemy imported from flask_sqlalchemy
Line: 9
pylint: reimported / Reimport 'db' (imported line 8)
pyflakes: F811 / redefinition of unused 'db' from line 8 (col 1)
Line: 27
pycodestyle: E305 / expected 2 blank lines after class or function definition, found 1 (col 1)
Line: 29
pylint: useless-object-inheritance / Class 'Config' inherits from object, can be safely removed from bases in python3
Line: 32
pycodestyle: E305 / expected 2 blank lines after class or function definition, found 1 (col 1)
Line: 39
pylint: no-value-for-parameter / No value for argument 'rss_urls' in function call (col 16)
pylint: no-value-for-parameter / No value for argument 'keywords' in function call (col 16)
Line: 46
pycodestyle: E305 / expected 2 blank lines after class or function definition, found 1 (col 1)
config.py
Line: 2
dodgy: secret / Possible hardcoded secret key
func.py
Line: 4
pylint: unused-import / Unused ceil imported from math
Line: 9
pylint: unused-argument / Unused argument 'articles_per_page' (col 40)
Line: 27
pylint: logging-fstring-interpolation / Use lazy % formatting in logging functions (col 4)
Line: 28
pylint: logging-fstring-interpolation / Use lazy % formatting in logging functions (col 4)
Line: 55
pylint: unspecified-encoding / Using open without explicitly specifying an encoding (col 13)
Line: 67
pylint: trailing-newlines / Trailing newlines
Ln 2, Col 30 (15 selected) Spaces: 4 UTF-8 LF Python 3.10.8 64-bit


File Edit Selection View Go Run Terminal Help
Assignment
EXPLORER
OPEN EDITORS 1 unsaved
func.py
config.py
ASSIGNMENT
__pycache__
config.cpython-39.pyc
database.cpython-39.pyc
func.cpython-39.pyc
models.cpython-39.pyc
routes.cpython-39.pyc
instance
news.db
templates
add_news.html
home.html
login.html
monitoring.html
news.html
profile.html
register.html
uploads
app.py
application.log
config.py
CS437 Assignment 1.docx
database.py
func.py
itu.xml
OUTLINE
TIMELINE
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Python
pylint: global-variable-not-assigned / Using global for 'news_storage' but no assignment is done (col 8)
PS C:\Users\Berkant\Downloads\Assignment\Assignment> prospector
Messages
========
app.py
Line: 2
pylint: unused-import / Unused SQLAlchemy imported from flask_sqlalchemy
Line: 9
pylint: reimported / Reimport 'db' (imported line 8)
pyflakes: F811 / redefinition of unused 'db' from line 8 (col 1)
Line: 27
pycodestyle: E305 / expected 2 blank lines after class or function definition, found 1 (col 1)
Line: 29
pylint: useless-object-inheritance / Class 'Config' inherits from object, can be safely removed from bases in python3
Line: 32
pycodestyle: E305 / expected 2 blank lines after class or function definition, found 1 (col 1)
Line: 39
pylint: no-value-for-parameter / No value for argument 'rss_urls' in function call (col 16)
pylint: no-value-for-parameter / No value for argument 'keywords' in function call (col 16)
Line: 46
pycodestyle: E305 / expected 2 blank lines after class or function definition, found 1 (col 1)
config.py
Line: 2
dodgy: secret / Possible hardcoded secret key
func.py
Line: 4
pylint: unused-import / Unused ceil imported from math
Line: 9
pylint: unused-argument / Unused argument 'articles_per_page' (col 40)
Line: 27
pylint: logging-fstring-interpolation / Use lazy % formatting in logging functions (col 4)
Line: 28
pylint: logging-fstring-interpolation / Use lazy % formatting in logging functions (col 4)
Line: 55
pylint: unspecified-encoding / Using open without explicitly specifying an encoding (col 13)
Line: 67
pylint: trailing-newlines / Trailing newlines
Ln 2, Col 30 (15 selected) Spaces: 4 UTF-8 LF Python 3.10.8 64-bit





RATS(Rough-Auditing-Tool-for-Security):

- **OS:** Linux (Since the RATS is a tool that mainly designed for Linux, it hass been decided to conduct the tests with RATS on Linux.)
- **Findings:**
    - Bad Token
    - File not Found
    - Duration of the analyse

- **Conclusion:**
    - Eventhough RATS offers different, results such as duration for timeout measurements and more, but It was not capable of bringing up findings that are related to CWE-315 & XXE. So It would be better to move on with a different analyser for this task.

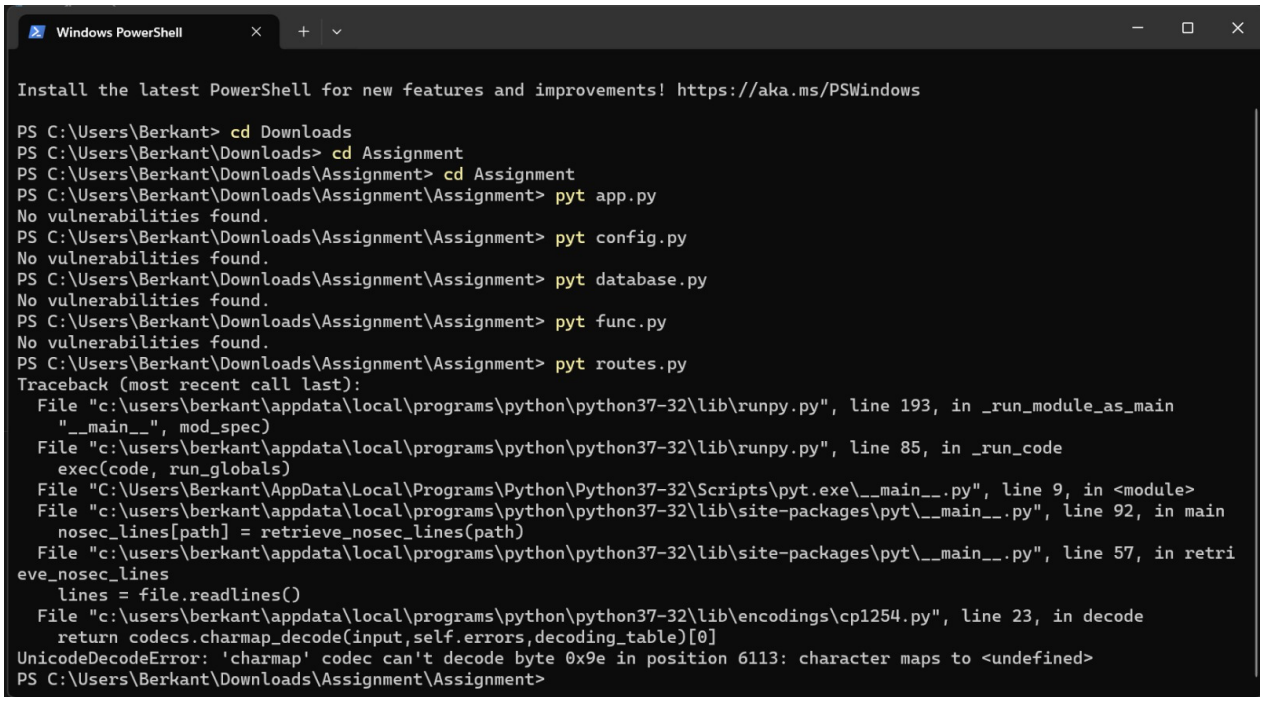
- Installation and usage process of RATS was easy, but it should be mentioned in the documentation that the easiest way to use the program is, using it on a LINUX operated system.

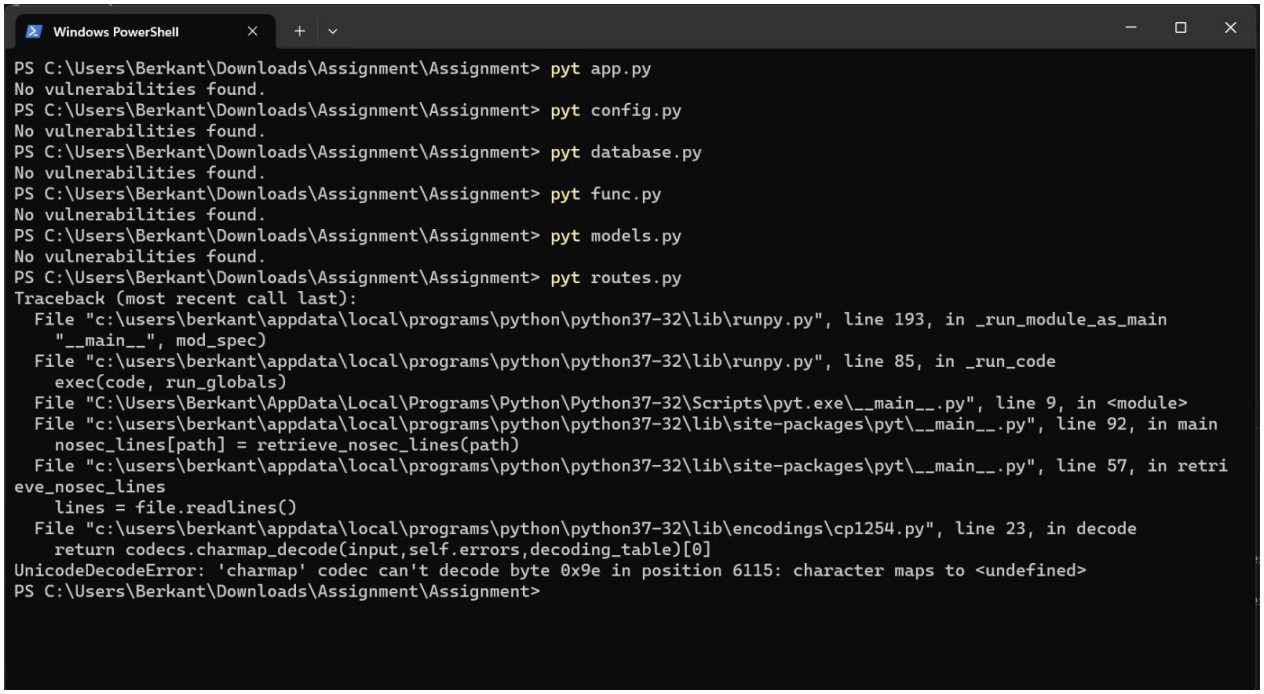
- **Screenshots:**

PYT (Python Taint):

- **OS:** Windows
- **Findings:**
  - Analyse made by PYT resulted as ”No vulnerabilities found.” throughout the whole project or made it scan python files one by one.
- **Conclusion:**
  - It can be concluded that the PYT(python-taint) did not work properly, since it is already known that project contains some vulnerabilities. So it can be said that the results given by PYT are not correct, another code analyser would be more propriate for the project.
  - Installation and usage of the program was challenging due to lack of information with the documentation and the program is not appropriate to use with latet versions of the required programs, versions of the programs should be rearrenged to make the program functioning.

- **Screenshots:**


Windows PowerShell
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows
PS C:\Users\Berkant> cd Downloads
PS C:\Users\Berkant\Downloads> cd Assignment
PS C:\Users\Berkant\Downloads\Assignment> cd Assignment
PS C:\Users\Berkant\Downloads\Assignment\Assignment> pyt app.py
No vulnerabilities found.
PS C:\Users\Berkant\Downloads\Assignment\Assignment> pyt config.py
No vulnerabilities found.
PS C:\Users\Berkant\Downloads\Assignment\Assignment> pyt database.py
No vulnerabilities found.
PS C:\Users\Berkant\Downloads\Assignment\Assignment> pyt func.py
No vulnerabilities found.
PS C:\Users\Berkant\Downloads\Assignment\Assignment> pyt routes.py
Traceback (most recent call last):
File "c:\users\berkant\appdata\local\programs\python\python37-32\lib\runpy.py", line 193, in _run_module_as_main
"__main__", mod_spec)
File "c:\users\berkant\appdata\local\programs\python\python37-32\lib\runpy.py", line 85, in _run_code
exec(code, run_globals)
File "C:\Users\Berkant\AppData\Local\Programs\Python\Python37-32\Scripts\pyt.exe__main__.py", line 9, in <module>
File "c:\users\berkant\appdata\local\programs\python\python37-32\lib\site-packages\pyt__main__.py", line 92, in main
nosec_lines[path] = retrieve_nosec_lines(path)
File "c:\users\berkant\appdata\local\programs\python\python37-32\lib\site-packages\pyt__main__.py", line 57, in retri
eve_nosec_lines
lines = file.readlines()
File "c:\users\berkant\appdata\local\programs\python\python37-32\lib\encodings\cp1254.py", line 23, in decode
return codecs.charmap_decode(input,self.errors,decoding_table)[0]
UnicodeDecodeError: 'charmap' codec can't decode byte 0x9e in position 6113: character maps to <undefined>
PS C:\Users\Berkant\Downloads\Assignment\Assignment>


Windows PowerShell
PS C:\Users\Berkant\Downloads\Assignment\Assignment> pyt app.py
No vulnerabilities found.
PS C:\Users\Berkant\Downloads\Assignment\Assignment> pyt config.py
No vulnerabilities found.
PS C:\Users\Berkant\Downloads\Assignment\Assignment> pyt database.py
No vulnerabilities found.
PS C:\Users\Berkant\Downloads\Assignment\Assignment> pyt func.py
No vulnerabilities found.
PS C:\Users\Berkant\Downloads\Assignment\Assignment> pyt models.py
No vulnerabilities found.
PS C:\Users\Berkant\Downloads\Assignment\Assignment> pyt routes.py
Traceback (most recent call last):
File "c:\users\berkant\appdata\local\programs\python\python37-32\lib\runpy.py", line 193, in _run_module_as_main
"__main__", mod_spec)
File "c:\users\berkant\appdata\local\programs\python\python37-32\lib\runpy.py", line 85, in _run_code
exec(code, run_globals)
File "C:\Users\Berkant\AppData\Local\Programs\Python\Python37-32\Scripts\pyt.exe__main__.py", line 9, in <module>
File "c:\users\berkant\appdata\local\programs\python\python37-32\lib\site-packages\pyt__main__.py", line 92, in main
nosec_lines[path] = retrieve_nosec_lines(path)
File "c:\users\berkant\appdata\local\programs\python\python37-32\lib\site-packages\pyt__main__.py", line 57, in retri
eve_nosec_lines
lines = file.readlines()
File "c:\users\berkant\appdata\local\programs\python\python37-32\lib\encodings\cp1254.py", line 23, in decode
return codecs.charmap_decode(input,self.errors,decoding_table)[0]
UnicodeDecodeError: 'charmap' codec can't decode byte 0x9e in position 6115: character maps to <undefined>
PS C:\Users\Berkant\Downloads\Assignment\Assignment>

Roles and Responsibilities:

Yağız: Developed the entire codebase for the Flask application, including routes, user authentication, database interactions, and XML parsing functionalities. Integrated basic security measures and intentionally introduced vulnerabilities for demonstration purposes and documented the code and provided comments for clarity and future reference.
Berkant: I installed the mentioned tools and run the source code in the given tools, Analysed the results provided by the tools. Then I inspected the results and decided wether the tool and given result approiorate for finding the vulnerability that we implemented. I mentioned whether the tool is suitible for finding the searched vulnerability and reported them. I also commented on whether the given tool is easy to install and use or not. Finally after the functioning project developped by Yağız, I tried to fix some bugs and misfunctionings.

Cenk: In the project, in addition to the given static code analysis tools for Python - Bandit, PYT (Python Taint), Rough-Auditing-Tool-for-Security, and Prospector - I independently identified and tested the Flake8 tool. This involved analyzing our Python codebase with Flake8 to detect the vulnerability. I installed the tool, made the analysis and gave an explanation about the anlysis.