Максимальное количество баллов за олимпиаду — 100

Задание 1.

B2B SaaS

Файлы: `2_9_01-windows.exe`, `2_9_01-linux`.

Давным-давно в далёкой-далёкой галактике менеджеры всего мира пытались найти метрику, позволяющую оценить объём работы программистов. Кто-то считал строчки кода, кто-то — количество коммитов, кто-то — количество токенов. Но недавно они нашли вариант лучше: B2B SaaS-проекты. И наш герой как раз попал на повторное техническое интервью в жёлтой компании. Ему поставили непосильную задачу: собрать 50000 B2B SaaS-проектов! Помогите ему в этом, иначе он будет уволен.

Формат ответа: `vsosh{text}`.

Ответ: `vsosh{v1b3_c0d3d_my_w4y_0ut}`

Критерий оценивания: точное совпадение ответа — 10 баллов

Максимальный балл за задание — 10

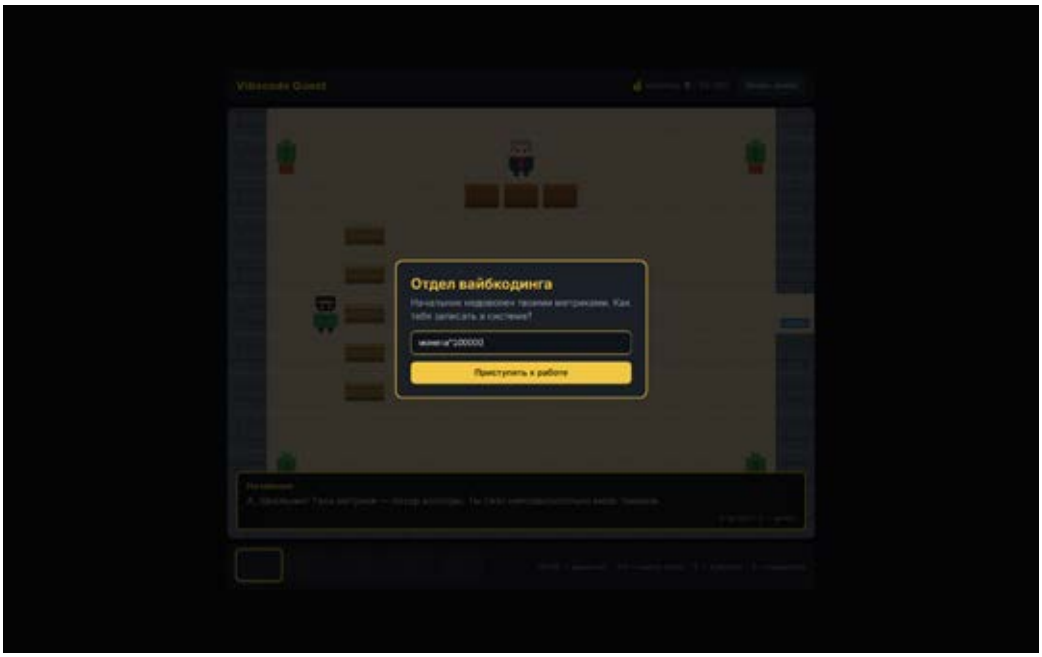
Решение.

Запускаем выданный файл (`2_9_01-windows.exe` для Windows или `2_9_01-linux` для Linux). Открывается браузер со страницей игры «Vibecode Quest». Вводим любое имя, например «Тестер», и нажимаем «Приступить к работе». В офисе находятся Начальник и Приёмщик; счётчик монет показывает 0/50000.

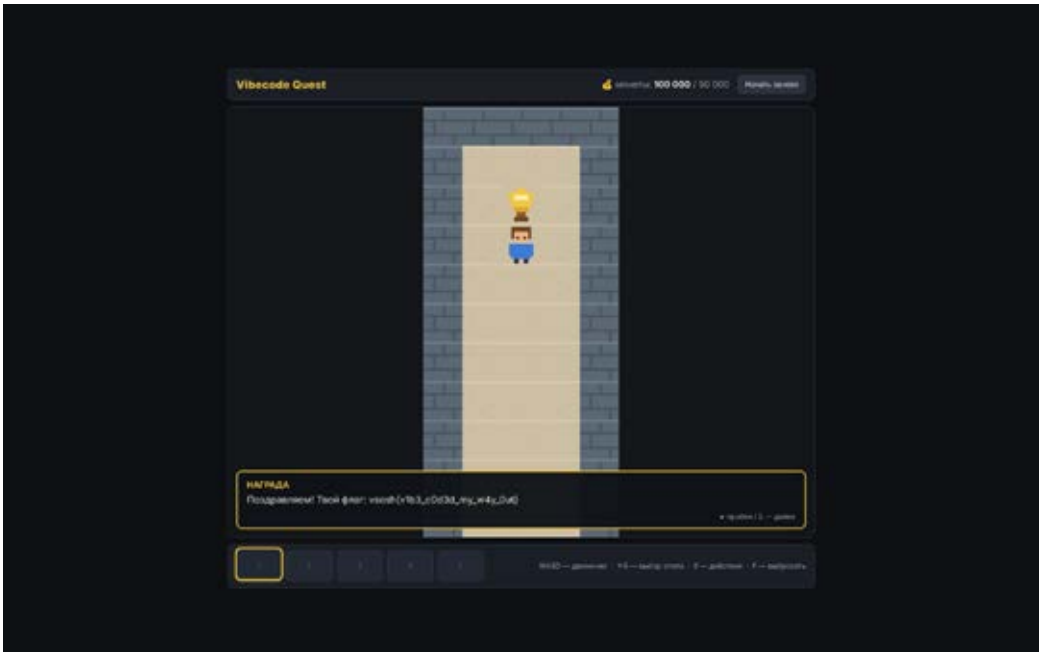
Подходим к Начальнику (клавиша E). Ему нужно 50000 монет за B2B SaaS-проекты. В соседней комнате взаимодействуем с компьютером и получаем токен. Отдав токен крабу и подождав 15 секунд, получаем проект. Сдаём его Приёмщику за одну монету. Такой способ получения 50000 монет потребовал бы больше восьми суток непрерывной игры.

В репликах перед начислением награды встречается запись вида `токен^1`, `проект^1`, `монета^1`. В инструментах разработчика видим, что действия отправляют запрос `POST /api/action`, а реплика Начальника включает введённое игроком имя без проверки. Текст реплики затем разбирается программой как команда на начисление награды.

Нажимаем «Начать заново» и вводим имя `монета^100000`. После разговора с Начальником эта строка оказывается в его реплике; счётчик показывает 100000/50000, и турникет открывается.



Проходим к пьедесталу с наградой. Там появляется флаг `vsosh{v1b3_c0d3d_my_w4y_0ut}`.



Уязвимость заключается в подстановке имени игрока в реплику, которую программа затем обрабатывает как команду начисления награды.

Задание 2.

Мемы динозавров

Файлы: `2_9_02-windows.exe`, `2_9_02-linux`.

Доктора наук института мемологии проводят исследования в области истории мемов, и они наткнулись на очень древний артефакт. Стряхнув пыль с главной страницы, они увидели то, что было забыто этим миром, — демотиваторы! Проанализировав сигнатуры сайта, было выявлено, что на его сервере в корне диска скрывается файл `secret` с ключом ко всем демотиваторам, но навыков учёных мемологии не хватило, чтобы найти его. Помогите им в этом.

Формат ответа: `vsosh{text}`.

Ответ: `vsosh{Y0u_f0un7_m3}`

Критерий оценивания: точное совпадение ответа — 10 баллов

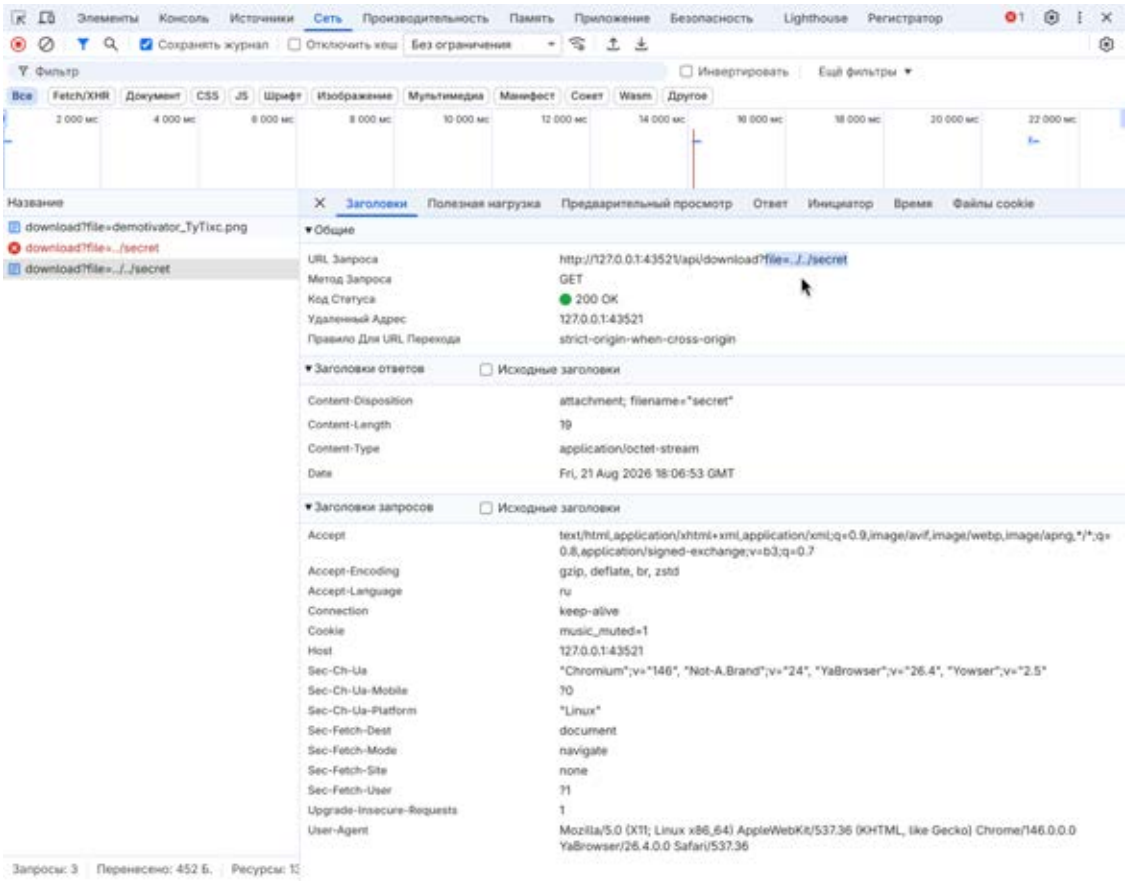
Максимальный балл за задание — 10

Решение.

Запускаем выданный файл для своей операционной системы. В браузере открывается генератор демотиваторов. После создания демотиватора сайт предлагает ссылку для его скачивания, например `http://127.0.0.1:PORT/api/download?fil`

Параметр `file` задаёт имя файла на сервере. В инструментах разработчика (вкладка «Сеть») виден запрос `GET /api/download?file=...`. Чтобы проверить, допускается ли выход из каталога с демотиваторами, используем `../`. Обращение к `http://127.0.0.1:PORT/api/download?file=../secret` даёт ошибку 404; добавляем ещё один переход к родительскому каталогу: `http://127.0.0.1:PORT/api/download?file=../../secret`. Вместо `PORT` используем порт из ссылки на скачивание.

Теперь сервер отдаёт файл `secret`, в котором содержится `vsosh{Y0u_f0un7_m3}`.



Ошибка состоит в том, что сервер не проверяет значение параметра `file` и позволяет выйти за пределы каталога с демотиваторами.

Задание 3.

Шифр наоборот

Файл: `encrypt_text.txt`. Рекомендуемые инструменты: Python.

Для передачи секретных сообщений была написана программа на языке Python:

```python
ALPHABET = (
    "abcdefghijklmnopqrstuvwxyz"
    "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
    "0123456789{}_"
)

def encrypt(text):
    result = ""
    for i, char in enumerate(text):
        pos = ALPHABET.index(char)

        new_pos = (pos + 3 + i) % len(ALPHABET)

        result += ALPHABET[new_pos]
    return result

flag = "??????????????????"
print(encrypt(flag))

```

В результате программа вывела: `ywtyofAoGqEGtptAIBzNv`.

Найдите исходное сообщение, представляющее осмысленную фразу.

Формат ответа: `vsosh{text}`.

Ответ: `vsosh{reverse_cipher}`

Критерий оценивания: точное совпадение ответа — 10 баллов

Максимальный балл за задание — 10

Решение.

При шифровании к позиции символа в заданном алфавите прибавляется $3 + i$. Для расшифровки выполняем обратное действие: `pos = new_pos - 3 - i`. С учётом заданного алфавита код дешифровки имеет вид:

```
ALPHABET = (
    "abcdefghijklmnopqrstuvwxyz"
    "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
    "0123456789{}_"
)

def decrypt(text):
    result = ""
    for i, char in enumerate(text):
        pos = ALPHABET.index(char)
        old_pos = (pos - 3 - i) % len(ALPHABET)
        result += ALPHABET[old_pos]
    return result
```

Применив этот код к строке из условия, получаем `vsosh{reverse_cipher}`.

Задание 4.

Удалённый кадр

Рекомендуемые инструменты: Autopsy. Файл: `2_9_04-снимок.img`.

В ходе расследования была изъята USB-флешка, вам передан её образ. Известно, что владелец флешки удалил некоторые данные перед её изъятием. Кроме того, следователь сообщил следующую информацию: «Интересующий расследование снимок был сделан 18 августа 2026 года примерно в 17:40». Помогите следователю найти скрытую информацию.

Формат ответа: `vsosh{text}`.

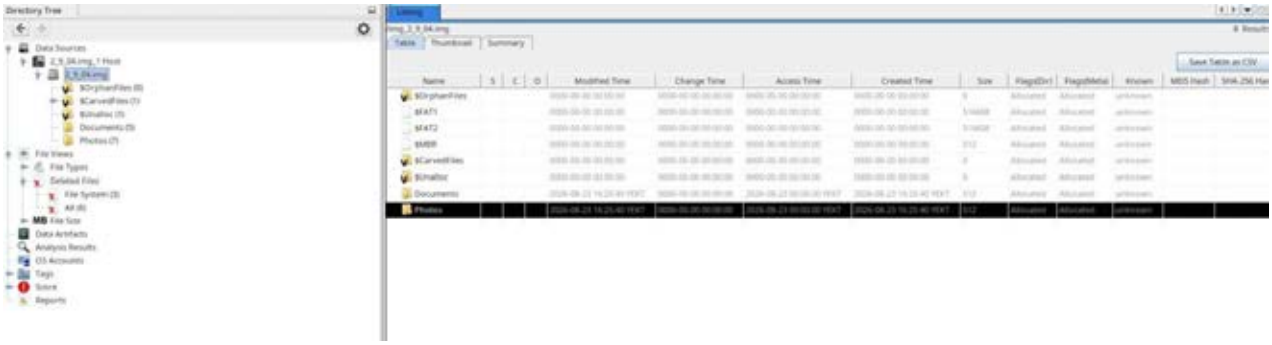
Ответ: `vsosh{deleted_files_are_not_gone}`

Критерий оценивания: точное совпадение ответа — 10 баллов

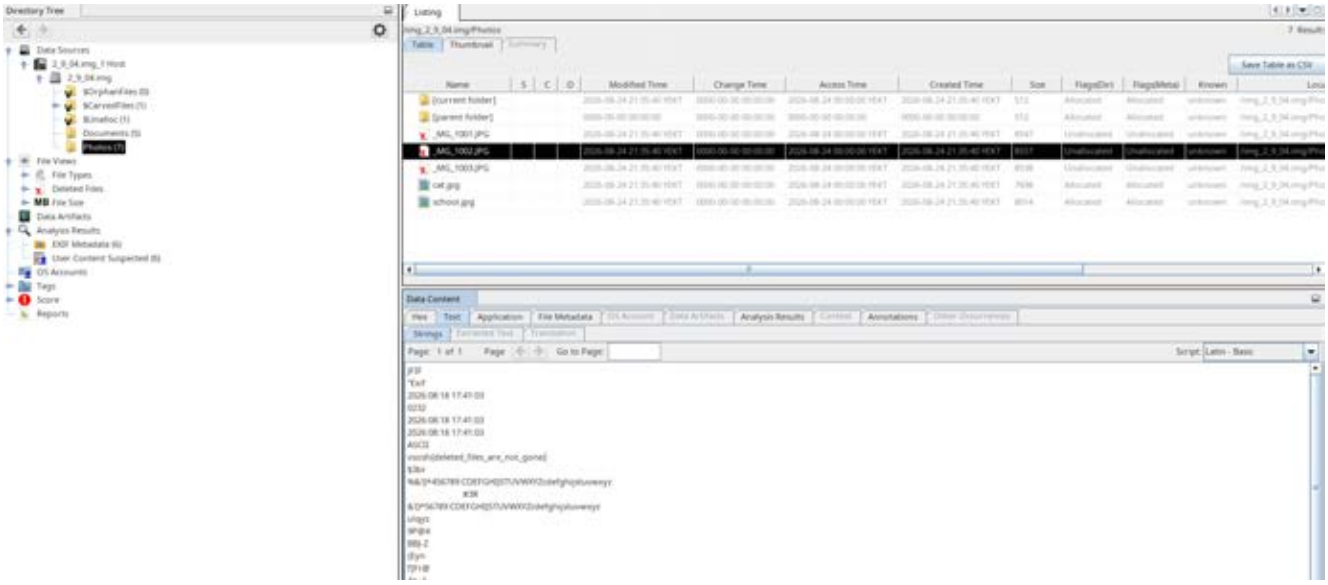
Максимальный балл за задание — 10

Решение.

Открываем образ диска в Autopsy. На диске есть каталоги `Photos` и `Documents`. Поскольку требуется найти снимок, изучаем `Photos`.



В каталоге видны три удалённых изображения. Изучаем их метаданные во вкладке `Text` окна `Data Content`. Снимок `_MG_1002.JPG` был сделан 18 августа 2026 года в 17:41:03; в его метаданных находится флаг `vsosh{deleted_files_are_not_gone}`.



Задание 5.

Сканирование сервера

Рекомендуемые инструменты: Wireshark. Файл: `2_9_05_traffic.pcap`.

Администратор школьной сети заметил подозрительную активность: один из компьютеров начал обращаться к другому компьютеру на большое количество разных TCP-портов. Администратор сохранил фрагмент сетевого трафика в файл `2_9_05_traffic.pcap`. Проанализируйте трафик и определите:

- какой компьютер выполнял сканирование;
- какие TCP-порты доступны на целевом компьютере;
- какой из обнаруженных сервисов может использоваться для удалённого управления компьютером.

Для анализа рекомендуется использовать программу Wireshark. В условии дана таблица портов и сервисов:

Порт	Сервис
21	FTP
22	SSH
23	Telnet
25	SMTP
53	DNS
80	HTTP
110	POP3
443	HTTPS
445	SMB
3389	RDP

а) Какой IP-адрес принадлежит компьютеру, с которого выполнялось сканирование?

Ответ: `192.168.20.50`

б) Какие TCP-порты, судя по полученным ответам, были доступны на целевом компьютере? Выберите из приведённых в таблице.

Ответ: 21, 80, 443, 3389

в) Какой из обнаруженных сервисов обычно используется для удалённого управления компьютером с помощью графического интерфейса?

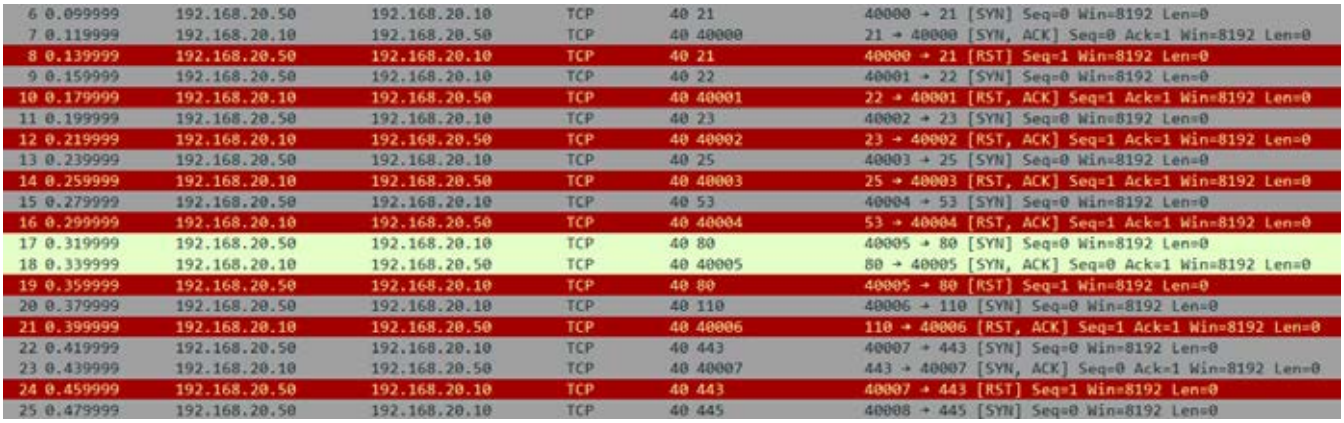
Ответ: RDP (также принимаются Remote Desktop Protocol, Remote Desktop)

Критерий оценивания: точное совпадение ответа — 10 баллов

Максимальный балл за задание — 10

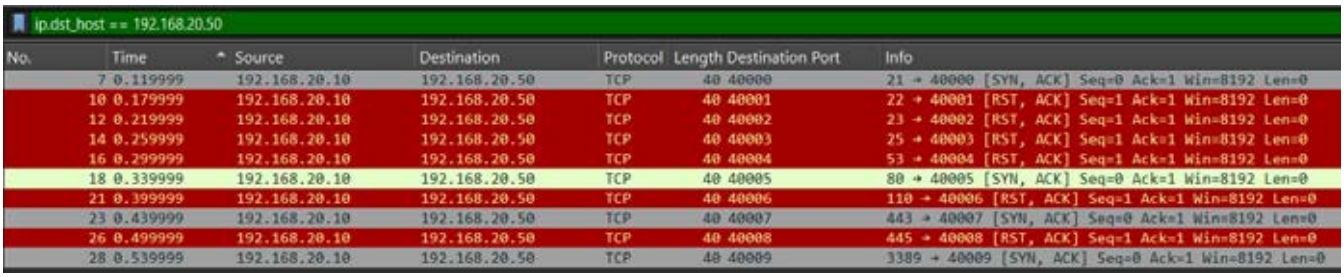
Решение.

Изучаем трафик: компьютер `192.168.20.50` отправляет множество TCP-пакетов на разные порты компьютера `192.168.20.10`. Следовательно, сканирование выполняет `192.168.20.50`.



6 0.099999	192.168.20.50	192.168.20.10	TCP	40 21	40000 → 21 [SYN] Seq=0 Win=8192 Len=0
7 0.119999	192.168.20.10	192.168.20.50	TCP	40 40000	21 → 40000 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0
8 0.139999	192.168.20.50	192.168.20.10	TCP	40 21	40000 → 21 [RST] Seq=1 Win=8192 Len=0
9 0.159999	192.168.20.50	192.168.20.10	TCP	40 22	40001 → 22 [SYN] Seq=0 Win=8192 Len=0
10 0.179999	192.168.20.10	192.168.20.50	TCP	40 40001	22 → 40001 [RST, ACK] Seq=1 Ack=1 Win=8192 Len=0
11 0.199999	192.168.20.50	192.168.20.10	TCP	40 23	40002 → 23 [SYN] Seq=0 Win=8192 Len=0
12 0.219999	192.168.20.10	192.168.20.50	TCP	40 40002	23 → 40002 [RST, ACK] Seq=1 Ack=1 Win=8192 Len=0
13 0.239999	192.168.20.50	192.168.20.10	TCP	40 25	40003 → 25 [SYN] Seq=0 Win=8192 Len=0
14 0.259999	192.168.20.10	192.168.20.50	TCP	40 40003	25 → 40003 [RST, ACK] Seq=1 Ack=1 Win=8192 Len=0
15 0.279999	192.168.20.50	192.168.20.10	TCP	40 53	40004 → 53 [SYN] Seq=0 Win=8192 Len=0
16 0.299999	192.168.20.10	192.168.20.50	TCP	40 40004	53 → 40004 [RST, ACK] Seq=1 Ack=1 Win=8192 Len=0
17 0.319999	192.168.20.50	192.168.20.10	TCP	40 80	40005 → 80 [SYN] Seq=0 Win=8192 Len=0
18 0.339999	192.168.20.10	192.168.20.50	TCP	40 40005	80 → 40005 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0
19 0.359999	192.168.20.50	192.168.20.10	TCP	40 80	40005 → 80 [RST] Seq=1 Win=8192 Len=0
20 0.379999	192.168.20.50	192.168.20.10	TCP	40 110	40006 → 110 [SYN] Seq=0 Win=8192 Len=0
21 0.399999	192.168.20.10	192.168.20.50	TCP	40 40006	110 → 40006 [RST, ACK] Seq=1 Ack=1 Win=8192 Len=0
22 0.419999	192.168.20.50	192.168.20.10	TCP	40 443	40007 → 443 [SYN] Seq=0 Win=8192 Len=0
23 0.439999	192.168.20.10	192.168.20.50	TCP	40 40007	443 → 40007 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0
24 0.459999	192.168.20.50	192.168.20.10	TCP	40 443	40007 → 443 [RST] Seq=1 Win=8192 Len=0
25 0.479999	192.168.20.50	192.168.20.10	TCP	40 445	40008 → 445 [SYN] Seq=0 Win=8192 Len=0

Фильтруем ответы сервера по адресу назначения `192.168.20.50`. Ответ с флагами SYN, ACK означает, что сервер принимает подключение; RST, ACK означает, что порт закрыт.



ip.dst_host == 192.168.20.50

No.	Time	Source	Destination	Protocol	Length	Destination Port	Info
7	0.119999	192.168.20.10	192.168.20.50	TCP	40	40000	21 → 40000 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0
10	0.179999	192.168.20.10	192.168.20.50	TCP	40	40001	22 → 40001 [RST, ACK] Seq=1 Ack=1 Win=8192 Len=0
12	0.219999	192.168.20.10	192.168.20.50	TCP	40	40002	23 → 40002 [RST, ACK] Seq=1 Ack=1 Win=8192 Len=0
14	0.259999	192.168.20.10	192.168.20.50	TCP	40	40003	25 → 40003 [RST, ACK] Seq=1 Ack=1 Win=8192 Len=0
16	0.299999	192.168.20.10	192.168.20.50	TCP	40	40004	53 → 40004 [RST, ACK] Seq=1 Ack=1 Win=8192 Len=0
18	0.339999	192.168.20.10	192.168.20.50	TCP	40	40005	80 → 40005 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0
21	0.399999	192.168.20.10	192.168.20.50	TCP	40	40006	110 → 40006 [RST, ACK] Seq=1 Ack=1 Win=8192 Len=0
23	0.439999	192.168.20.10	192.168.20.50	TCP	40	40007	443 → 40007 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0
26	0.499999	192.168.20.10	192.168.20.50	TCP	40	40008	445 → 40008 [RST, ACK] Seq=1 Ack=1 Win=8192 Len=0
28	0.539999	192.168.20.10	192.168.20.50	TCP	40	40009	3389 → 40009 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0

По полученным ответам составляем таблицу:

Порт	Ответ	Состояние
21	SYN/ACK	открыт
22	RST/ACK	закрыт
23	RST/ACK	закрыт
25	RST/ACK	закрыт
53	RST/ACK	закрыт
80	SYN/ACK	открыт
110	RST/ACK	закрыт
443	SYN/ACK	открыт
445	RST/ACK	закрыт
3389	SYN/ACK	открыт

Порт 3389 используется протоколом RDP (Remote Desktop Protocol), который предназначен для удалённого подключения к рабочему столу компьютера.

Задание 6.

Чем опасен этот код?

Разработчики учебного веб-приложения оставили несколько фрагментов кода, которые могут быть небезопасными. Специалист по информационной безопасности обнаружил, что злоумышленник может передавать приложению собственные данные. В зависимости от того, как программа использует эти данные, последствия могут быть разными.

Установите соответствие между фрагментами кода и последствиями, которые могут произойти из-за ошибок в них. Каждый фрагмент соединяется ровно с одним последствием, и каждое последствие используется ровно один раз.

Ответ:

Фрагмент кода	Последствие

A.
`user_id = request.args.get("id")`
`query = "SELECT * FROM users WHERE id = "+ user_id`
`db.execute(query)`
	Злоумышленник может изменить структуру запроса к базе данных и получить или изменить данные, доступ к которым ему не предоставлялся (SQL Injection).

B.
`name = request.args.get("name")`
`return «h1>Здравствуйте, "+ name + «/h1>"`
	Злоумышленник может заставить браузер посетителя выполнить JavaScript, который был передан через веб-сайт (Cross-Site Scripting).

Фрагмент кода	Последствие

C.
`filename = request.args.get("file")`
`path = "/home/site/files/"+ filename`
`with open(path, "r") as file:`
`return file.read()`
	Злоумышленник может получить содержимое файлов, находящихся за пределами разрешённой папки (Path Traversal).

D.
`host = request.args.get("host")`
`os.system("ping -c 1 "+ host)`
	Злоумышленник может заставить приложение выполнить команды операционной системы, которые разработчик не предусмотрел (OS Command Injection).

E.
`expression =`
`request.args.get("expression")`
`result = eval(expression)`
	Злоумышленник может заставить программу выполнить произвольные инструкции на языке Python (Python Code Injection).

Критерий оценивания: за каждую верную пару — 2 балла

Максимальный балл за задание — 10

Решение.

Прослеживаем, куда поступают данные, контролируемые пользователем.

Во фрагменте A значение `user_id` добавляется непосредственно к SQL-запросу. Пользователь может изменить его структуру и получить или изменить данные в базе (SQL Injection).

Во фрагменте B значение `name` без обработки вставляется в HTML. Переданный пользователем JavaScript может выполниться в браузере посетителя (Cross-Site Scripting).

Во фрагменте C значение `filename` становится частью пути к файлу. Последовательность `../` позволяет перейти в родительский каталог и получить доступ к файлам за пределами разрешённой папки (Path Traversal).

Во фрагменте D значение `host` добавляется к команде `ping`. Специальные символы командной оболочки позволяют исполнить дополнительные команды операционной системы (OS Command Injection).

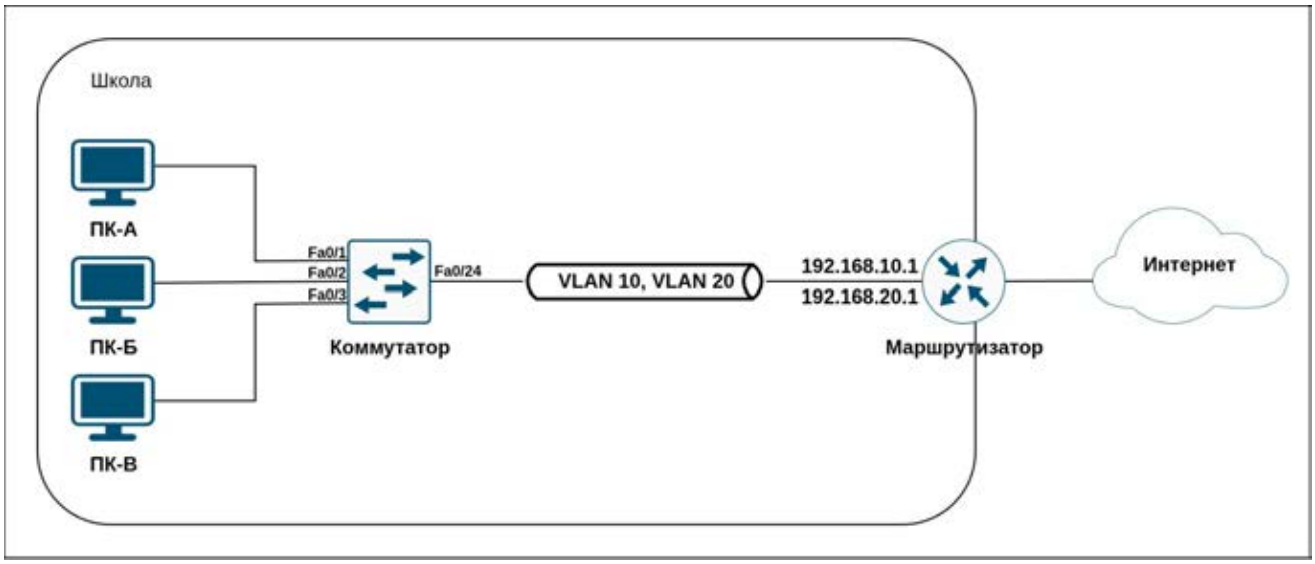
Во фрагменте E функция `eval()` интерпретирует переданную пользователем строку как выражение Python. Пользователь может повлиять на исполняемые программой инструкции (Python Code Injection).

Итак, соответствие: A — SQL Injection, B — Cross-Site Scripting, C — Path Traversal, D — OS Command Injection, E — Python Code Injection.

Задание 7.

Разделение сети

В школе необходимо разделить сеть на две части: ученики и администрация. Для этого администратор использует виртуальные сети (VLAN). Схема сети:



Известно:

- ПК-А — компьютер ученика;
- ПК-Б — компьютер ученика;
- ПК-В — компьютер администрации;
- порт коммутатора Fa0/1 подключён к ПК-А;
- порт коммутатора Fa0/2 подключён к ПК-Б;
- порт коммутатора Fa0/3 подключён к ПК-В;
- порт коммутатора Fa0/24 подключён к маршрутизатору.

Используются две VLAN:

VLAN	Назначение	Сеть
10	Ученики	`192.168.10.0/24`
20	Администрация	`192.168.20.0/24`

а) С какими VLAN связаны порты коммутатора?

Ответ:

Порт	VLAN
Fa0/1	VLAN 10
Fa0/2	VLAN 10
Fa0/3	VLAN 20

б) Какой режим должен использовать порт Fa0/24, соединяющий коммутатор с маршрутизатором? Варианты: Access VLAN 10; Access VLAN 20; Trunk; Disabled.

Ответ: Trunk

в) Администратор подготовил варианты IP-адресов. Какой вариант соответствует схеме?

- ПК-А — `192.168.10.15`, ПК-В — `192.168.20.15`;
- ПК-А — `192.168.20.15`, ПК-В — `192.168.10.15`;
- ПК-А — `192.168.10.15`, ПК-В — `192.168.10.16`;
- ПК-А — `192.168.20.15`, ПК-В — `192.168.20.16`.

Ответ: ПК-А — `192.168.10.15`, ПК-В — `192.168.20.15`

г) После настройки сети администратор проверяет соединение: ПК-А — ПК-Б работает; ПК-А — ПК-В не работает; ПК-А — Интернет работает. Выберите верные утверждения:

- ПК-А и ПК-Б находятся в одной VLAN;
- ПК-А и ПК-В находятся в разных VLAN;
- если ПК-А выходит в Интернет, то VLAN 10 и VLAN 20 обязательно напрямую соединены;
- для обмена данными между VLAN необходима маршрутизация;
- VLAN 10 и VLAN 20 автоматически объединяются коммутатором.

Ответ: ПК-А и ПК-Б находятся в одной VLAN; ПК-А и ПК-В находятся в разных VLAN; для обмена данными между VLAN необходима маршрутизация.

Критерий оценивания: точное совпадение ответа — 10 баллов

Максимальный балл за задание — 10

Решение.

ПК-А и ПК-Б принадлежат ученикам, значит, их порты Fa0/1 и Fa0/2 относятся к VLAN 10. ПК-В принадлежит администрации, значит, порт Fa0/3 относится к VLAN 20.

Порты типа access используют для подключения конечных устройств. Порт Fa0/24 соединяет коммутатор с маршрутизатором и передаёт данные обеих VLAN, поэтому для него нужен режим trunk.

ПК-А должен иметь адрес из сети `192.168.10.0/24`, а ПК-В — из сети `192.168.20.0/24`. Условию соответствует первый вариант: `192.168.10.15` для ПК-А и `192.168.20.15` для ПК-В.

ПК-А и ПК-Б находятся в одной VLAN 10, а ПК-В — в VLAN 20. Выход ПК-А в Интернет не означает, что две VLAN напрямую соединены. Для обмена данными между разными VLAN нужна маршрутизация; сам коммутатор их автоматически не объединяет. Верны первое, второе и четвёртое утверждения.

Задание 8.

Кухонный комбайн

Рекомендуемые инструменты: StegSolve. Файлы: `2_9_08_part1.png`, `2_9_08_part2.png`.

Перед вами два изображения. По отдельности они не содержат ничего особенно интересного. Кухонный комбайн хорош тем, что позволяет получить результат из нескольких отдельных ингредиентов. Возможно, и здесь стоит посмотреть на изображения не по отдельности. Найдите скрытое сообщение и запишите его в формате `vsosh{text}`.

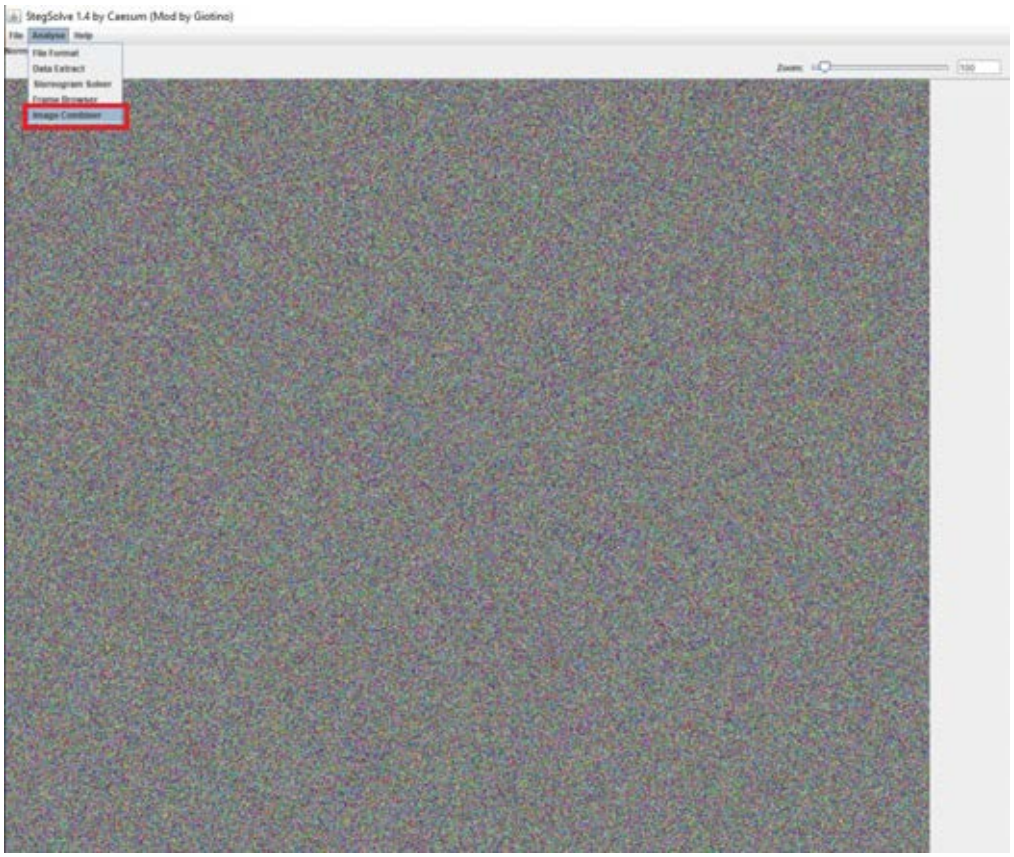
Ответ: `vsosh{p3rf3ct_c0mb1ne}`

Критерий оценивания: точное совпадение ответа — 10 баллов

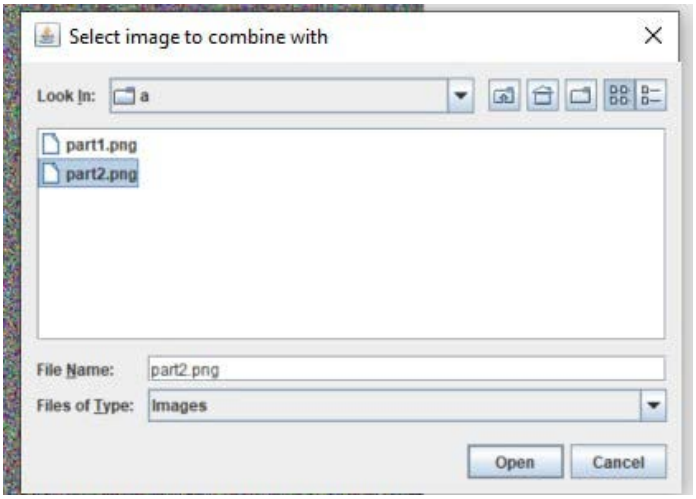
Максимальный балл за задание — 10

Решение.

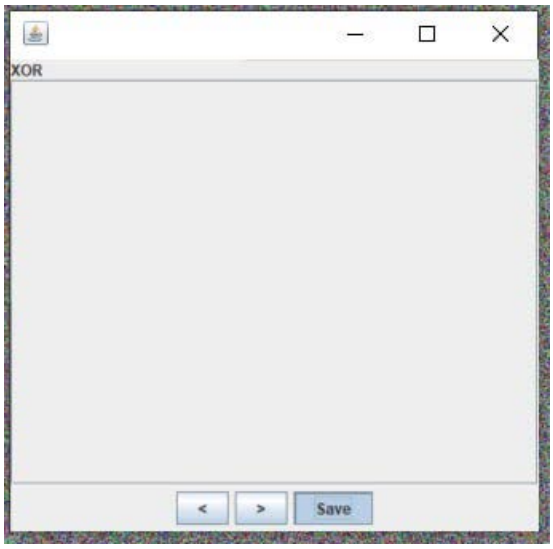
Каждое изображение по отдельности не даёт полезной информации. Формулировка задания подсказывает, что их следует совместить. Открываем первое изображение в StegSolve и выбираем меню `Analyse -- Image Combiner`.



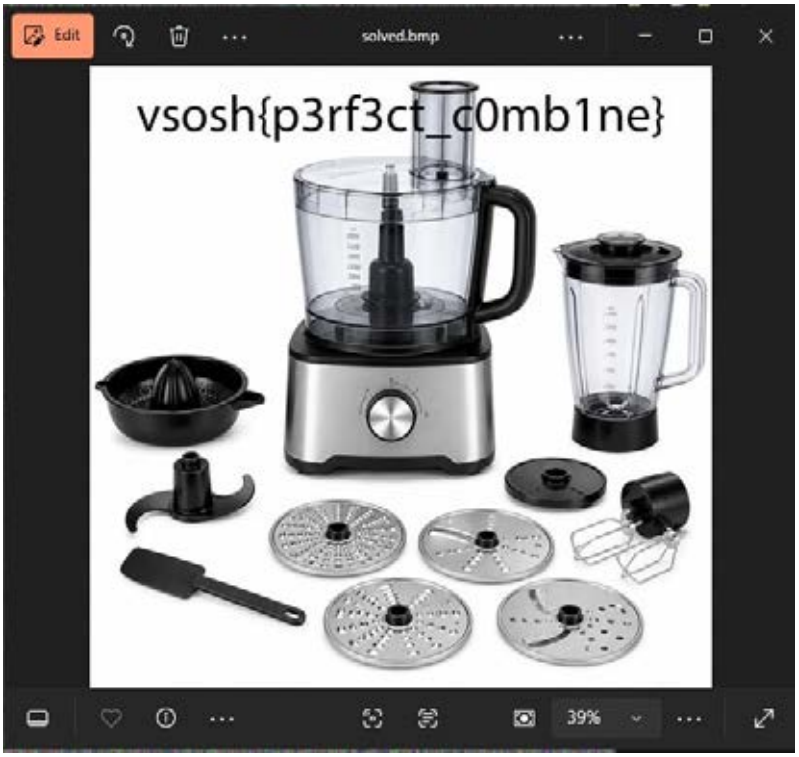
В окне `Image Combiner` выбираем второе изображение.



Выбираем режим XOR. Если результат не отображается в окне, сохраняем его и открываем полученный файл в просмотрщике изображений.



На совмещённом изображении читается `vsosh{p3rf3ct_c0mb1ne}`.



Задание 9.

Ненадёжный код

Рекомендуемые инструменты: Python. Файл: `2_9_09_code.py`.

Представленный листинг кода реализует веб-сервер для загрузки и просмотра пользовательских файлов. Изучите код и ответьте на вопросы ниже.

а) Что позволяет сделать злоумышленнику найденная уязвимость?

Ответ:

- Создать избыточную нагрузку на сервер
- ✓ Получить доступ к файловой системе сервера
- Получить возможность исполнять произвольный код на сервере
- Получить доступ во внутреннюю сеть, где находится сервер
- Получить список IP-адресов посетителей сайта

б) Какой текст должен присутствовать в запросе к серверу для успешной эксплуатации уязвимости?

Ответ:

- ✓ `../../../flag.txt`
- `"password": { "$gt": }`
- `(&(username=admin)(|(&)(password=hehe))`
- `` `"alg": "none" ``
- `<script>console.log("hacked")</script>`

в) Как исправить найденную уязвимость?

Ответ:

- Ограничить список разрешённых алгоритмов подписи токена
- Использовать экранирование данных при выводе на фронтенд
- ✓ Использовать нормализацию путей, полученных из пользовательского ввода
- Разрешить запросы только из внутренней сети
- Добавить капчу на страницу

Критерий оценивания: точное совпадение ответа — 10 баллов

Максимальный балл за задание — 10

Решение.

В обработчике `serve_file()` значение параметра `name` без проверки добавляется к пути для чтения файла:

```
file_name = request.args.get("name")
with open(f"assets/{file_name}", "r") as f:
    return f.read()
```

Последовательность `../` означает переход в родительский каталог. Поэтому сформированный путь может выйти за пределы `assets`. Злоумышленник может прочитать файлы, доступные процессу веб-сервера. Проверка расширения в `allowed_file()` применяется при загрузке файлов через `/upload`, а не при чтении через `/asset`.

В параметре `name` запроса `/asset` передаём `../../../flag.txt`. Программа попытается открыть `assets/../../../flag.t` Если рабочая директория веб-сервера — `/opt/filebin`, после нормализации этот путь указывает на `/flag.txt`. При наличии файла и праве доступа его содержимое вернётся в ответе сервера.

Для устранения уязвимости требуется нормализовать итоговый путь и проверить, что файл находится внутри разрешённой директории. В файле решений предлагается, например, использовать `Path.resolve()` и проверку вложенности полученного пути. Если путь выходит за границы директории, доступ к файлу нужно отклонить.

Задание 10.

Что запускать?

Специалист по информационной безопасности исследует учебный сервер. Установите соответствие между ситуациями и подходящими командами или утилитами. Каждый вариант можно использовать только один раз.

Ответ:

Ситуация	Подходящая команда или утилита
Нужно проверить доступность сервера по сети	`ping 192.168.10.15`
Нужно определить открытые TCP-порты на сервере	`nmap 192.168.10.15`
Нужно получить HTML-код главной страницы прямо из терминала, не открывая браузер	`curl http://192.168.10.15/`
Нужно понять, какое ПО веб-сервера отвечает на порту 80, и по возможности определить его версию	`nmap -sV -p 80 192.168.10.15`

Ситуация	Подходящая команда или утилита
Нужно проверить, соответствует ли MD5-хеш файла `secret.txt` значению `42f749ade7f9e195bf475f37a44cafcb`	`md5sum secret.txt`
Нужно посмотреть список файлов внутри архива `archive.zip`, но не извлекать их	`unzip -l archive.zip` Неиспользованные варианты: `chmod +x archive.zip`; `whoami`.

Критерий оценивания: за каждую верную пару — $\frac{5}{3}$ балла

Максимальный балл за задание — 10

Решение.

Для проверки сетевой доступности сервера используется утилита `ping`. Открытые TCP-порты определяют с помощью `nmap`. Получить HTML-код страницы из терминала позволяет `curl`.

Для определения ПО, отвечающего на порту 80, и его версии используется `nmap -sV -p 80`: параметр `-sV` определяет версию сервиса, `-p 80` ограничивает проверку портом 80.

Хеш файла вычисляется командой `md5sum secret.txt`. Список файлов ZIP-архива без извлечения показывает команда `unzip -l archive.zip`.

Команды `whoami` и `chmod +x archive.zip` не используются.

