**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Московский технический университет связи и информатики»**

Кафедра «Телевидения и звукового вещания им. С.И. Катаева (ТиЗВ)»

**Отчет по лабораторной работе № 3**

**по дисциплине**

**«Информационные технологии и программирование»**

**на тему**

**« Развертывание Docker контейнера Kali Linux для идентификации IP-камер и перебора URI видеопотоков»**

Студент: Косаткин Семен Владиславович

Группа: БИК2405

Проверил: Егоров Дмитрий Аркадьевич

Москва 2025 г.

**Цель работы**:

Получить практические навыки работы c Docker.

Получить практические навыки сканирования сети с помощью Nmap.

Разработать Bash скрипт для автоматического перебора URI видеопотоков.

Научиться использовать curl для анализа HTTP ответов.

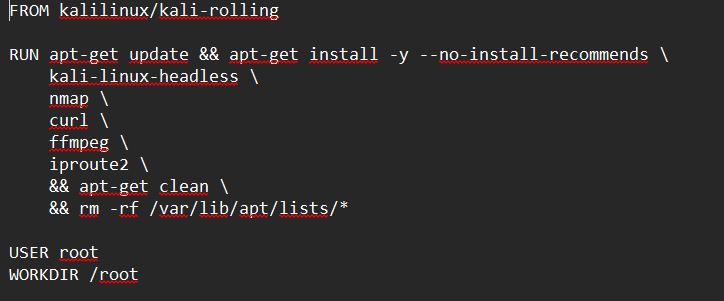
Научиться использовать ffmpeg для захвата кадров из видеопотока.

Приобрести опыт в документировании проделанной работы.

**Ход работы**

В ходе лабораторной работы были выполнены следующие этапы:

1.Создан Docker-образ на основе Kali Linux с предустановленными инструментами: nmap, curl, ffmpeg.



FROM kalilinux/kali-rolling: Базовый образ Kali Linux.

RUN apt-get update && apt-get install...: Обновляет пакетный менеджер и устанавливает необходимые инструменты.

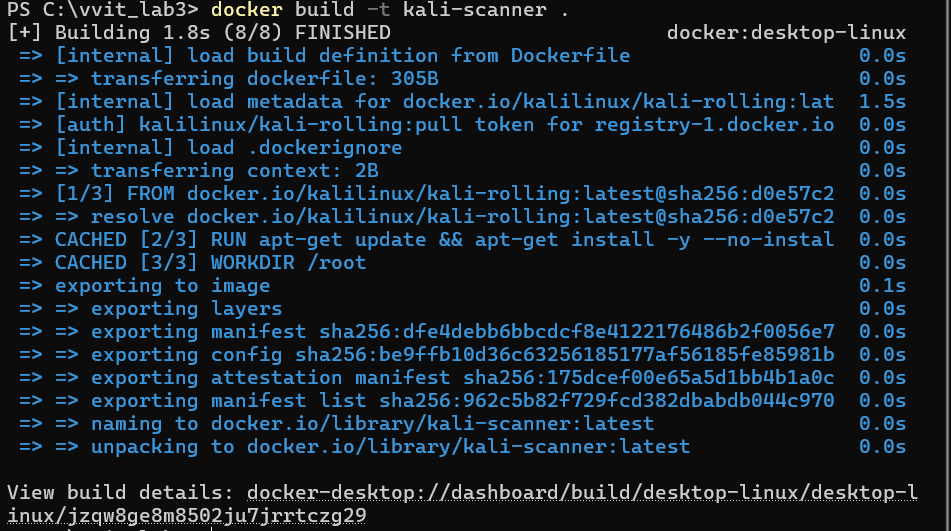
 --no-install-recommends уменьшает размер образа. iproute2 добавил для команды ip addr.

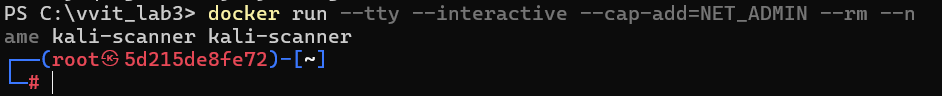
apt-get clean && rm -rf [/var/lib/apt/lists](https://colab.research.google.com/drive/1o1G09TUtvENV7gEspPUpsKKA9i5wM17H?usp=sharing&pli=1&authuser=1)/\*: Очищает кеш apt для уменьшения размера образа.

USER root: Возвращает пользователя к root (не рекомендуется для production, но упрощает задачу).

WORKDIR /root: Устанавливает рабочую директорию в /root.

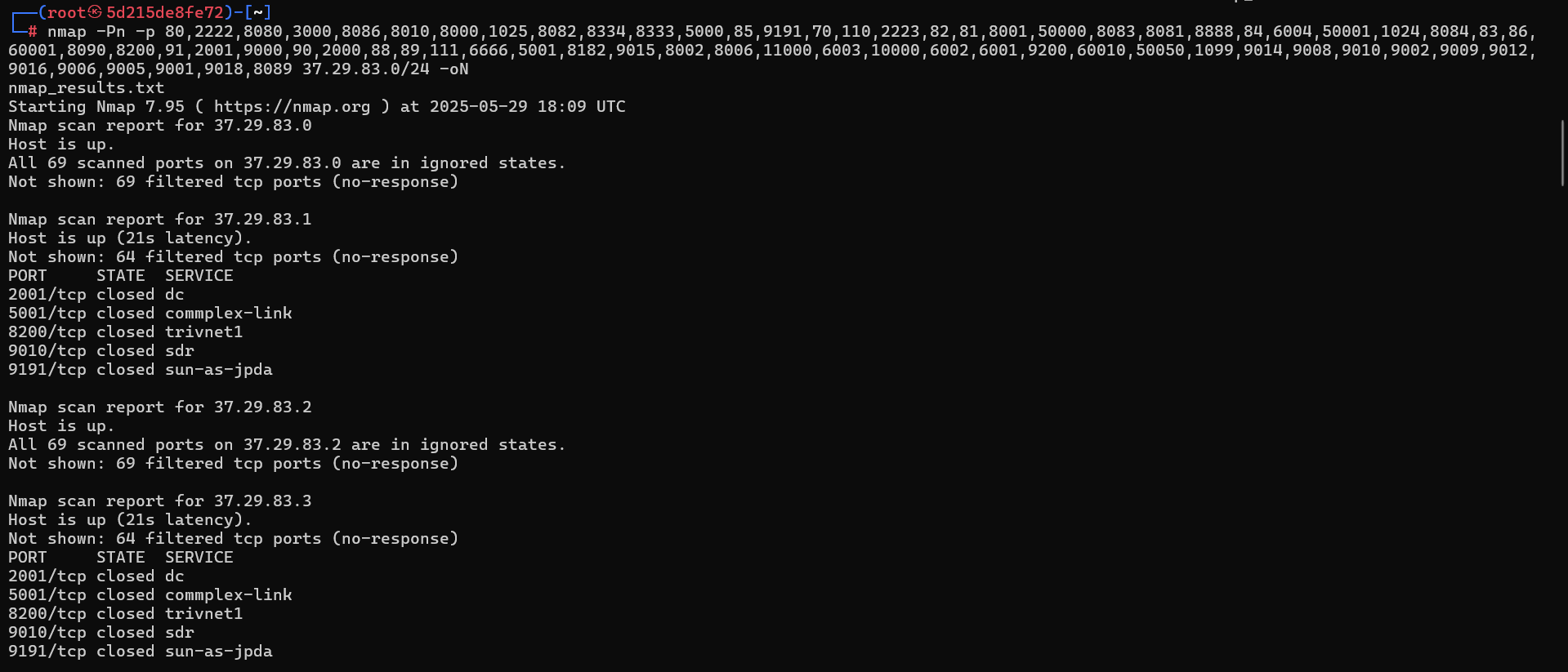
Сборка образа и запуск контейнера





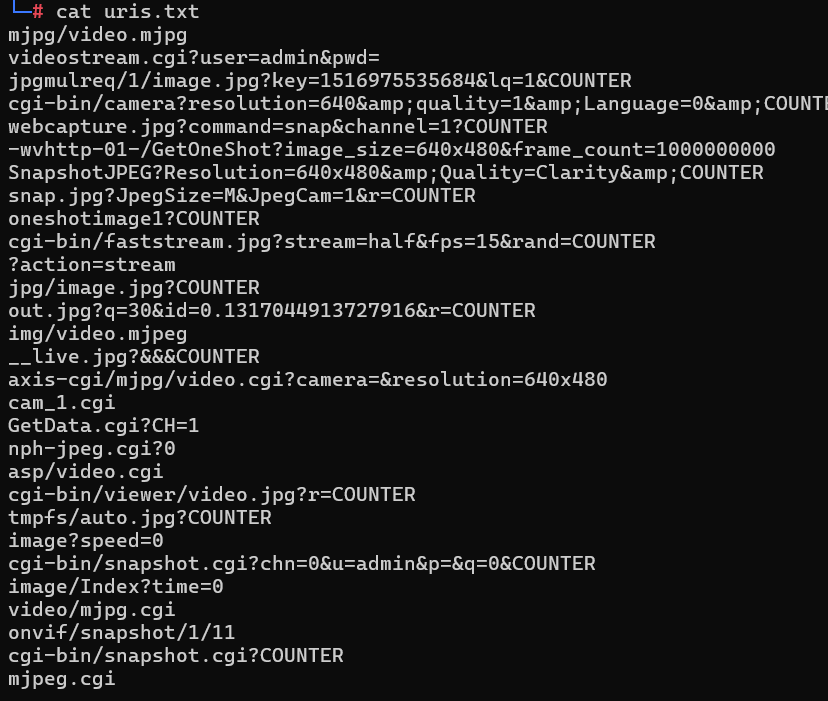
docker run --tty --interactive --cap-add=NET\_ADMIN --rm --name kali-scanner kali-scanner

* -it: Интерактивный терминал.
* --rm: Удаляет контейнер после остановки.
* --name: Задает имя контейнера.
* --cap-add=NET\_ADMIN: Дает права администрирования сетевых интерфейсов
* 2.Проведено сканирование сети с помощью nmap для обнаружения устройств с открытыми портами, характерными для IP-камер. Создан файл, который сохраняет в себя данные.
* с -p: Указывает список портов для сканирования.
* -oN: Сохраняет результаты в текстовый файл.
* -pN: Отключает предварительную ICMP проверку.



3.Разработан Bash-скрипт scan\_cams.sh для автоматического перебора URI видеопотоков на найденных IP-адресах.

Создание uris.txt с добавленными другими распространенными URI для IP-камер.

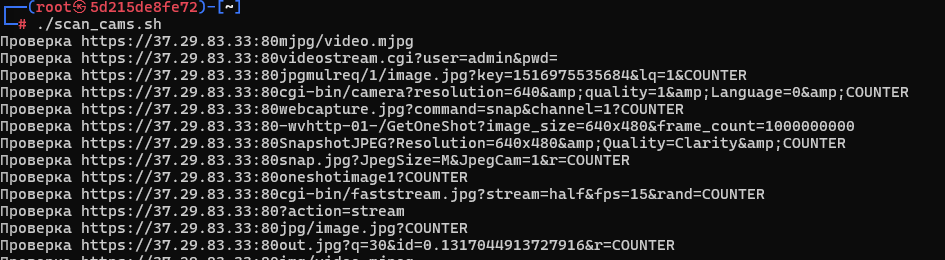




* curl -m 1 -Is: curl с параметрами:
  + -m 1: Таймаут 1 секунда.
  + -I: Запрашивает только заголовки HTTP.
  + -s: Тихий режим.
* grep "HTTP/": Проверяет, что получен HTTP-ответ. $? содержит код возврата последней команды.
* 200 OK, 401 Authorization Required и 501 Not Implemented: Наиболее вероятные коды ответов для IP-камер.

**Запуск скрипта:**

chmod +x scan\_cams.sh  
./scan\_cams.sh



С помощью сканирования была найдена изначальная сеть 37.29.83.0/24 ,но она не обслуживается и перебрасывает на другую сеть(рабочую).

4.Проверены HTTP-ответы с использованием curl, а также выполнен захват кадра из видеопотока с помощью ffmpeg. (При сканировании сети 37.24.94.0/24 были получены изображения с [http://37.24.94.66:80/](https://www.google.com/url?q=http%3A%2F%2F37.24.94.66%3A80%2F) при нескольких uri)

Используя в терминале команду ,

ffmpeg -i "http:// 37.24.94.66:80/mjpg/video.mjpg" -vframes 1 snapshot.jpg

делаю скриншот с камеры.



Все этапы задокументированы, файлы сохранены и проанализированы.

Выводы:

В ходе лабораторной работы я получил практические навыки работы c Docker;получила практические навыки сканирования сети с помощью Nmap; разработал Bash скрипт для автоматического перебора URI видеопотоков; научилcя использовать curl для анализа HTTP ответов; научился использовать ffmpeg для захвата кадров из видеопотока; приобрел опыт в документировании проделанной работы.

Данная лабораторная работа позволила получить практический опыт в использовании Docker для развертывания инструментов безопасности, а также навыки в сканировании сети, идентификации устройств и анализе веб-сервисов на предмет уязвимостей.

1.Безопасность IP-камер в сети

1) Их порты должны быть закрыты для публичного доступа, порт, а так же стандартные имя пользователя и пароль должны быть изменены (иначе, логин и пароль есть в общем доступе)

2) Открытые порты и сервисы

Камеры часто работают на стандартных портах (80, 554, 8080) с включенными сервисами, которые не требуют аутентификации.

3) Нешифрованные потоки

Многие устройства передают видео по незащищенным протоколам (HTTP вместо HTTPS, RTSP без шифрования).

4) Устаревшее ПО

Производители редко выпускают обновления прошивок, оставляя устройства уязвимыми к известным эксплойтам.

5) Доступ из интернета

Неправильная настройка NAT/PAT делает камеры доступными из всемирной сети.

2. Улучшения для скрипта scan\_cams.sh

1) Многопоточность (внесена), проверка на доступность (curl и ffprobe)

2) Поддержка RTSP/ONVIF

Включить проверку rtsp:// и onvif:// потоков.

3) Фильтрация ответов

Игнорировать ответы без Content-Type: image/jpeg.

4) Интеграция с Hydra

Автоматический подбор паролей при обнаружении кода 401.

5) Расширенное логирование

Сохранять результаты в JSON-формате для последующего анализа.

3. Альтернативные инструменты поиска

1) Shodan и censys search помогают определить сеть, ее провайдера, пройтись по адресам в этой сети, посмотреть открытые протоколы и наличие камер.

Поиск по ключевым словам: "webcam", "axis-cgi", "mjpg".

2)ONVIF Device Manager

Обнаружение совместимых устройств в локальной сети.

3)Angry IP Scanner

Быстрое сканирование диапазонов адресов.

4) iSpy (Windows)

Автоматическое обнаружение и управление камерами.

5)Metasploit Framework

Готовые модули для тестирования уязвимостей.

Рекомендации по безопасности

- Обязательная смена паролей по умолчанию

- Регулярное обновление прошивок

- Отключение неиспользуемых сервисов

- Использование VPN вместо проброса портов

- Ограничение доступа по MAC-адресам

Проведенная работа демонстрирует типовые уязвимости IP-камер и необходимость усиления их защиты. Разработанный инструментарий может быть использован для аудита безопасности, но требует доработки для промышленного применения. Все исследования проводятся только в разрешенных сетях.