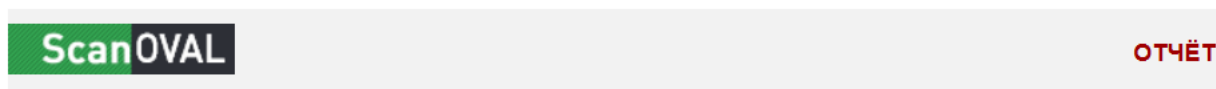


Задание №1

Я нашел 10 критических уязвимостей в основном связанных с OpenSSL.
77 высоких уязвимостей, 67 средних, 5 низких. Всего 159 уязвимостей.

Рекомендации по устранению – обновление ПО.

ОС достаточно защищена.



№ отчета	66f0ac0f-f28c-4726-ad57-762a6582b986
№ сканирования	bb468abc-b2dd-4273-84fa-7ff09ba408aa
Профиль	Уязвимости
Начало/завершение сканирования	12.02.2025 1:57:54 / 12.02.2025 2:04:13
Формирование отчета	12.02.2025 2:30:57

Уровень опасности	Найдено	Всего
Критический	10	5295
Высокий	77	10823
Средний	67	11258
Низкий	5	1231
Недоступно	0	1
Всего	159	28608

Задание №2

1) *Производитель оборудования:* Keenetic

2) *Операционная система, установленная на устройстве:* KeeneticOS

3) *Открытых портов на устройстве:* 7

4) *Доступные сервисы:*

23 порт – telnet

53 порт – DNS

80 порт – HTTP

139 порт – netbios ssn

443 порт – HTTPS

445 порт – Microsoft-ds

1900 порт – upnp

5) *Опасные сервисы:* telnet, http, dns

6) **Вывод:** при помощи данного теста утилитой nmap я выявил уязвимости своего домашнего роутера и устранил их, закрыв порты при помощи настроек межсетевого экрана telnet, http и сделав фильтрацию dns. Теперь порты telnet и http стоят в режиме «filtered».

```
C:\Users\user>nmap -p-
Starting Nmap 7.95 ( https://nmap.org ) at 2025-02-16 21:38 RTZ 6 (чшьр)
Nmap scan report for
Host is up (0.0064s latency).
Not shown: 65527 closed tcp ports (reset)
PORT      STATE      SERVICE
23/tcp    filtered  telnet
53/tcp    open      domain
80/tcp    filtered  http
139/tcp   open      netbios-ssn
443/tcp   open      https
445/tcp   open      microsoft-ds
1900/tcp  open      upnp
3702/tcp  open      ws-discovery
MAC Address: (Keenetic Limited)

Nmap done: 1 IP address (1 host up) scanned in 59.98 seconds
```

Задание №3

- 1) *Пакеты, которые генерируются в моей сети:* TCP, UDP, QUIC, TLS
- 2) *О содержимом пакетов я могу сказать следующее:*

Основные компоненты пакета в Wireshark:

1. **Заголовок канального уровня (Layer 2):**
 - **Ethernet Header:** Содержит MAC-адреса отправителя и получателя, а также тип протокола верхнего уровня.
2. **Заголовок сетевого уровня (Layer 3):**
 - **IP Header:** Включает IP-адреса источника и назначения, версию протокола (IPv4 или IPv6), время жизни пакета (TTL) и другую служебную информацию.
3. **Заголовок транспортного уровня (Layer 4):**
 - **TCP или UDP Header:** Содержит порты источника и назначения, номера последовательности (для TCP), флаги управления соединением и контрольные суммы.
4. **Данные прикладного уровня (Layer 7):**
 - **Payload:** Непосредственно данные, передаваемые приложениями, такие как содержимое HTTP-запросов и ответов, данные FTP, SMTP и других протоколов.

Анализ содержимого пакетов:

Wireshark предоставляет подробный анализ каждого уровня пакета, позволяя:

- **Рассматривать заголовки протоколов:** Детально изучать информацию, содержащуюся в заголовках различных протоколов, что помогает в диагностике сетевых проблем и понимании маршрутизации пакетов.
- **Анализировать данные приложений:** Просматривать и анализировать данные, передаваемые приложениями, что полезно для отладки и обеспечения безопасности.

- **Использовать фильтры:** Применять фильтры для отображения только интересующих пакетов или протоколов, что упрощает анализ больших объемов трафика.
- **Следить за потоками данных:** Объединять связанные пакеты в единые потоки для упрощения анализа сеансов связи, таких как TCP-сессии или HTTP-запросы и ответы.

3) *Мне встречалась множественное bad подключение по протоколу TCP*