Оглавление

Тема и цель работы

Тема лабораторной работы: анализ трафика сети посредством Wireshark

Цель работы: Научиться устанавливать, проводить базовые настройки и проверять работоспособность wireshark.

Вариант №25

Оборудование, ПО

Таблица 1 - Оборудование, ПО

Устройство	Операционная система	IP адрес/Маска	Шлюз	DNS
CLI_A1	Astra Linux SE 1.8.x	10.0.25.1/24	-	au-1.au.team.lab
CLI_A2	Astra Linux SE 1.8.x	10.0.25.2/24	-	au-2.au.team.lab
CLI_A3	Astra Linux SE 1.8.x	10.0.25.3/24	-	au-3.au.team.lab

Ход лабораторной работы

Открыть сетевой анализатор Wireshark командой **`wireshark`** (см. рисунок 1).

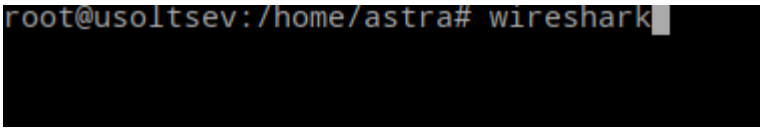

Рисунок 1 – включение Wireshark

Выбрать фильтр any для анализа любого трафика проходящего через машину (см. рисунок 2)

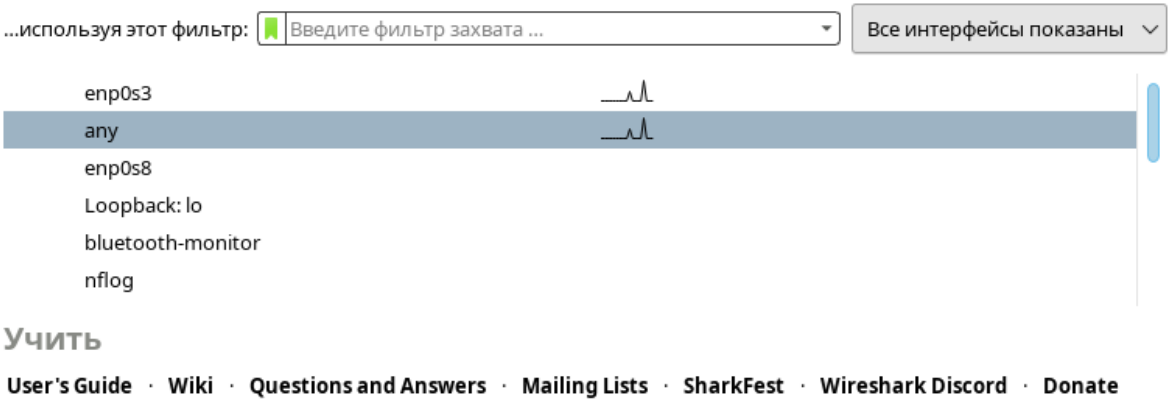

Рисунок 2 – фильтр any

Произвести пинг нашего доменного имени (см. рисунок 3).

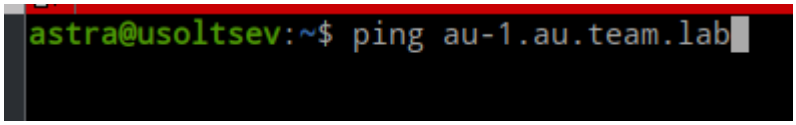

Рисунок 3 – пинг нашего доменного имени

Остановить захват трафика в wireshark (см. рисунок 4).

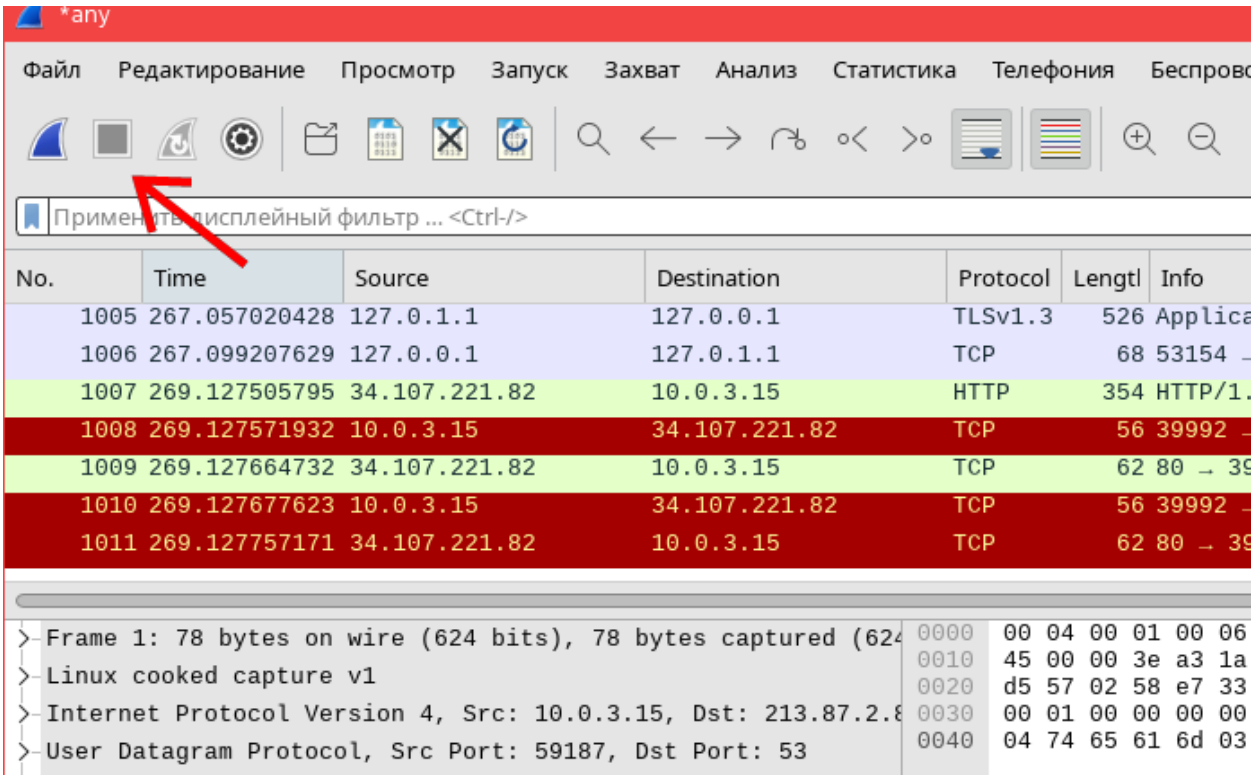

Рисунок 4 – остановка захвата трафика

Wireshark зафиксировал трафик исходящий от команды **`ping`** (см. рисунок 5).

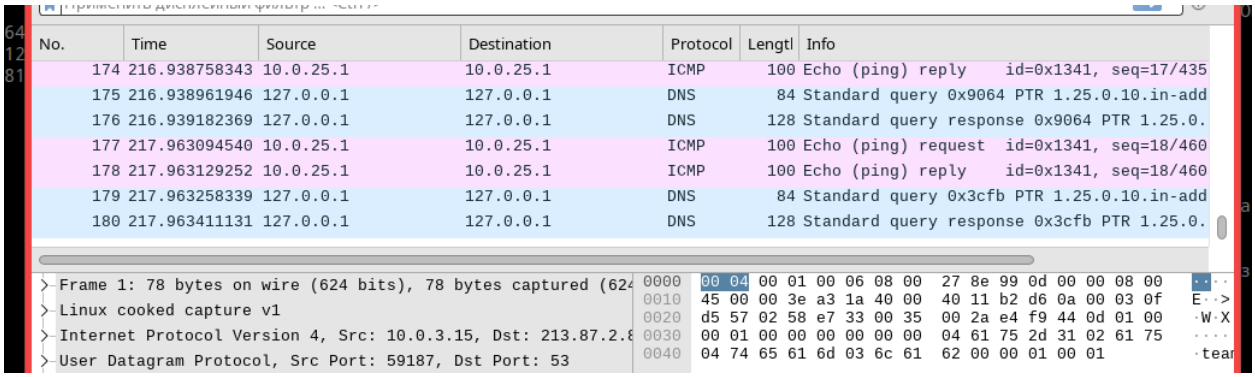

Рисунок 5 – команда ping в wireshark

Начать захват трафика и зайти на созданный ранее сайт astra. Остановить захват трафика. С помощью фильтра найдем весь трафик, отправленный с адреса 10.0.25.1 (ip.src==10.0.0.1) (см. рисунок 6).

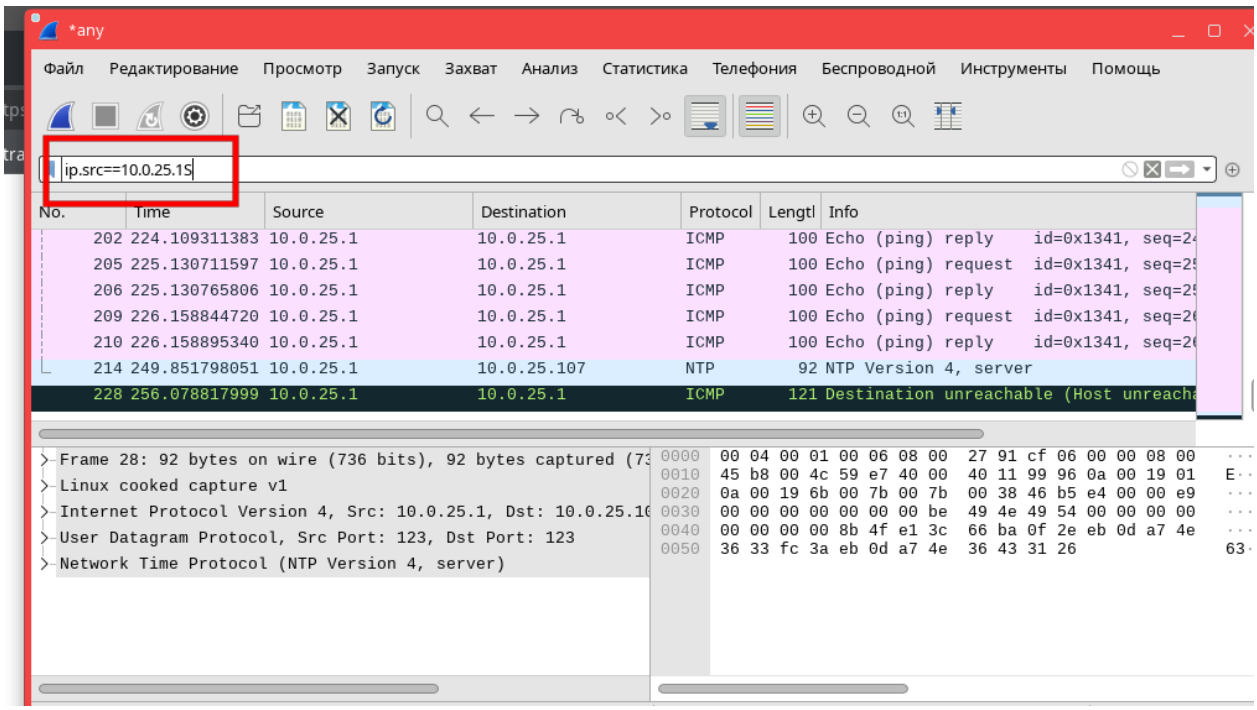

Рисунок 6 – применённый фильтр

С помощью фильтра найти весь трафик, где нет протокола ICMP: !icmp (см. рисунок 7)

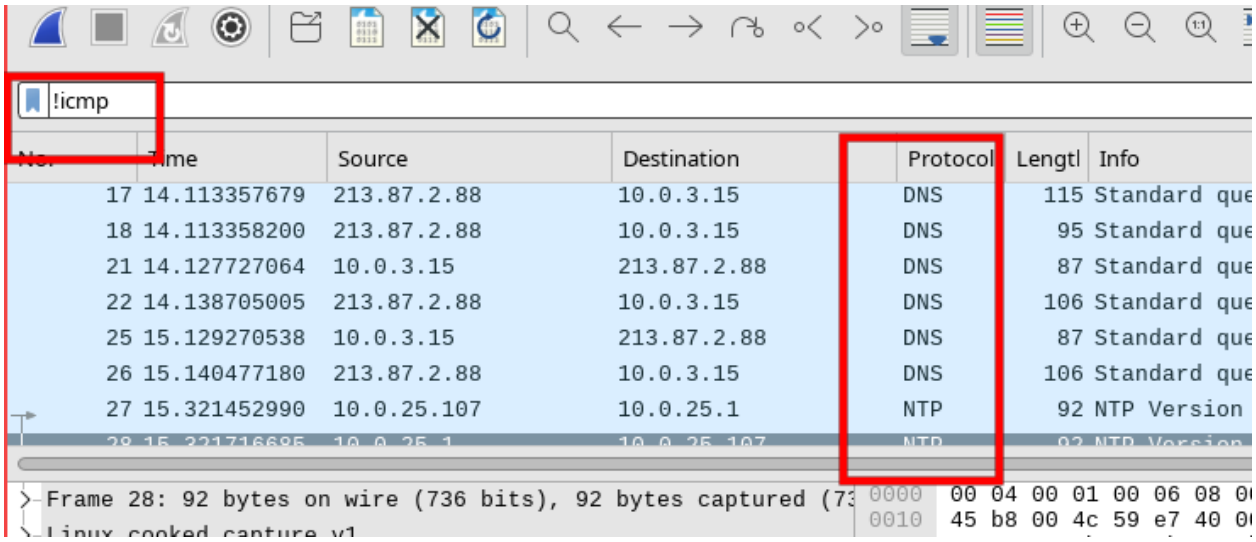

Рисунок 7 – фильтр !icmp

Эксперимент

На первой виртуальной машине (ВМ) открыть wireshark (см. рисунок 8)

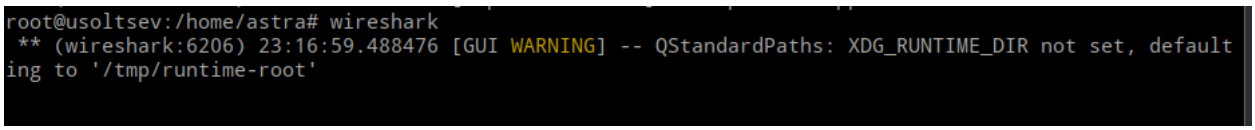

Рисунок 8 – открытие wireshark

Выбрать интерфейс enp0s3 для захвата трафика (см. рисунок 9).

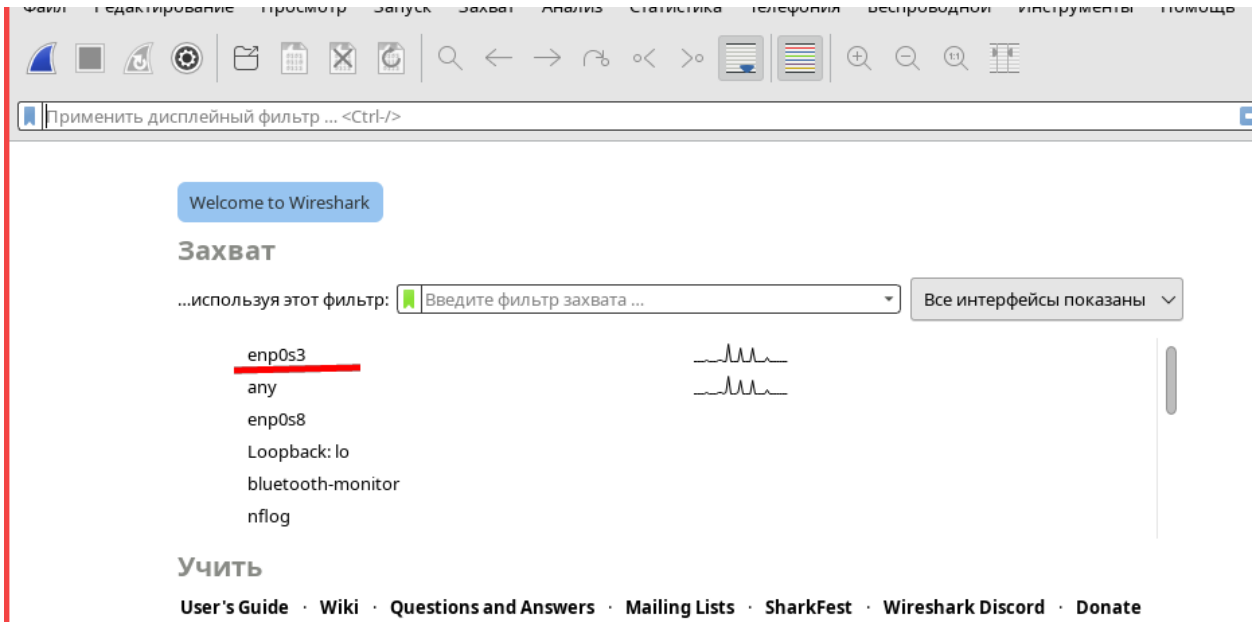

Рисунок 9 – фильтрация по интерфейсу enp0s3

На второй ВМ установить инструмент для сканирования портов Nmap командой **`apt-get install nmap`** (см. рисунок 10).

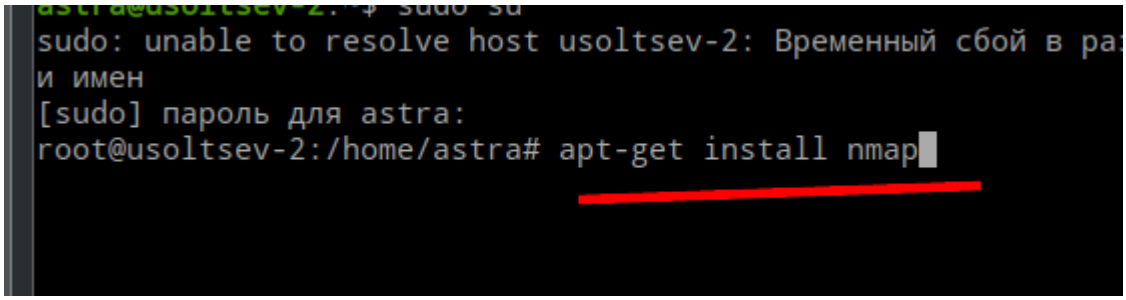

Рисунок 10 – установка сканера

Выполним команду сканирования портов SYN-сканированием первой ВМ командой **`nmap -sS 10.0.25.1`** (см. рисунок 11).

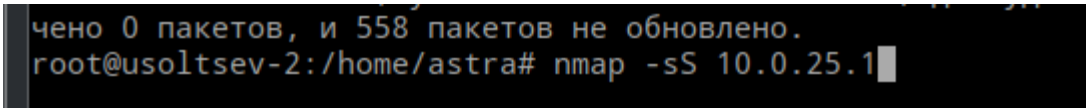

Рисунок 11 – сканирование портов

Команда показывает открытые на данный момент порты на первой ВМ (см. рисунок 12).

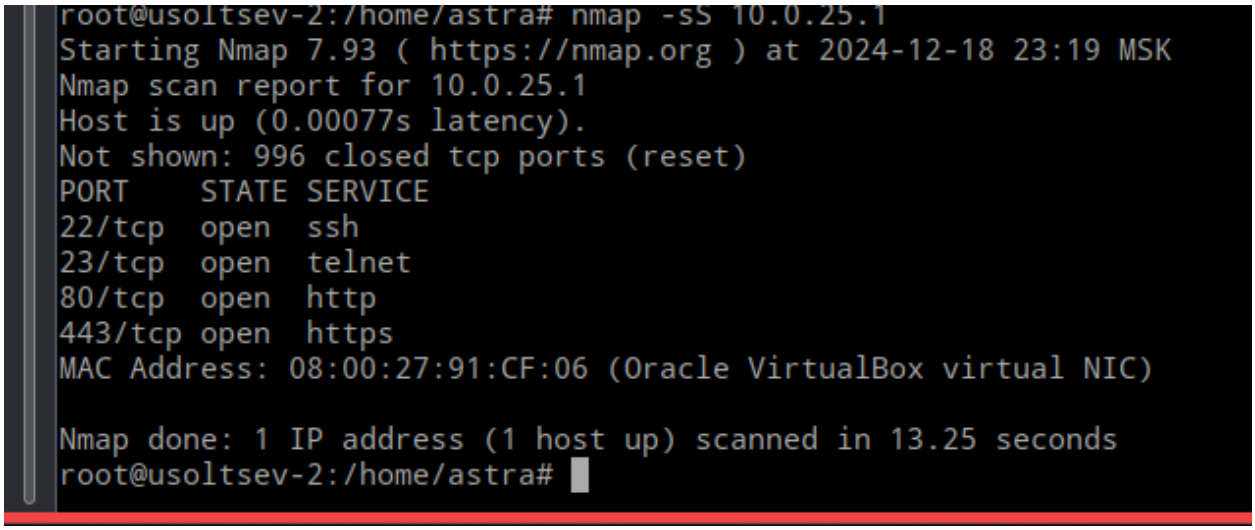

Рисунок 12 – открытые порты на первой ВМ

Остановить захват трафика (см. рисунок 13).

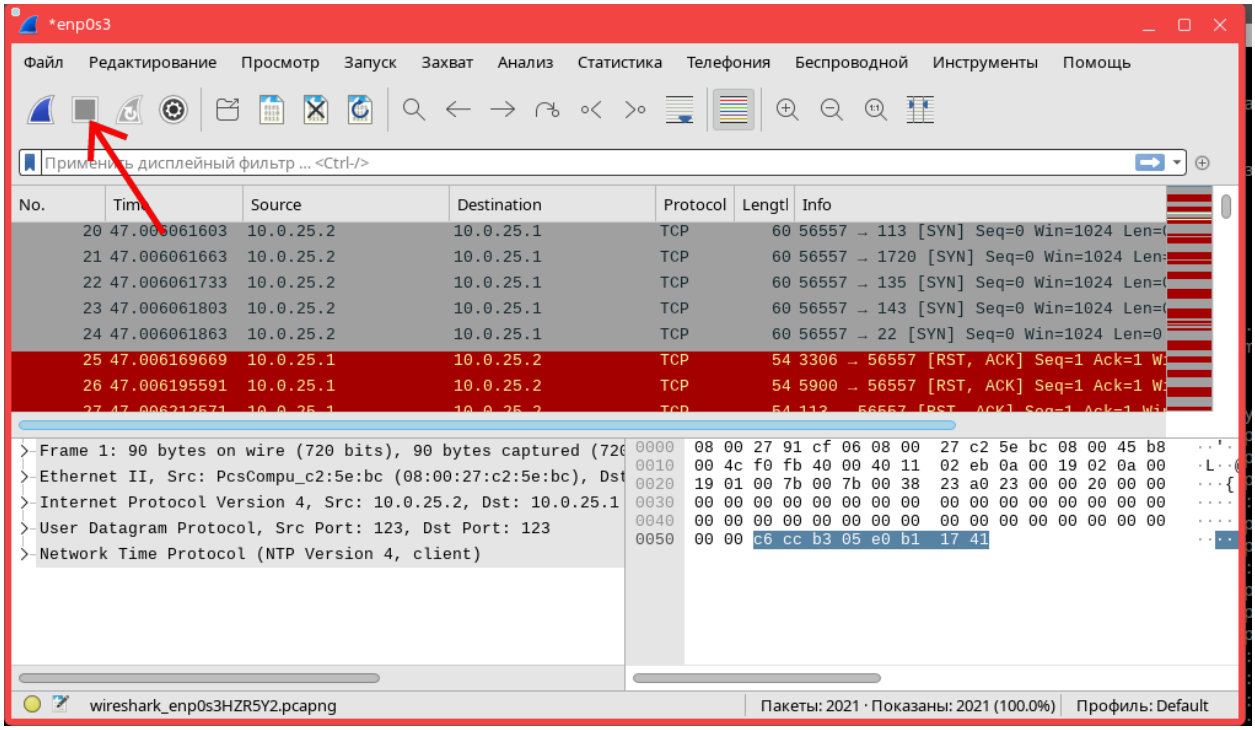

Рисунок 13 – остановка захвата трафика

В строке фильтра Wireshark ввести:

tcp.flags.syn == 1 && tcp.flags.ack == 0

Wireshark захватил 1000 пакетов протокола TCP. Именно такое количество портов просканировал Nmap (см. рисунок 14).

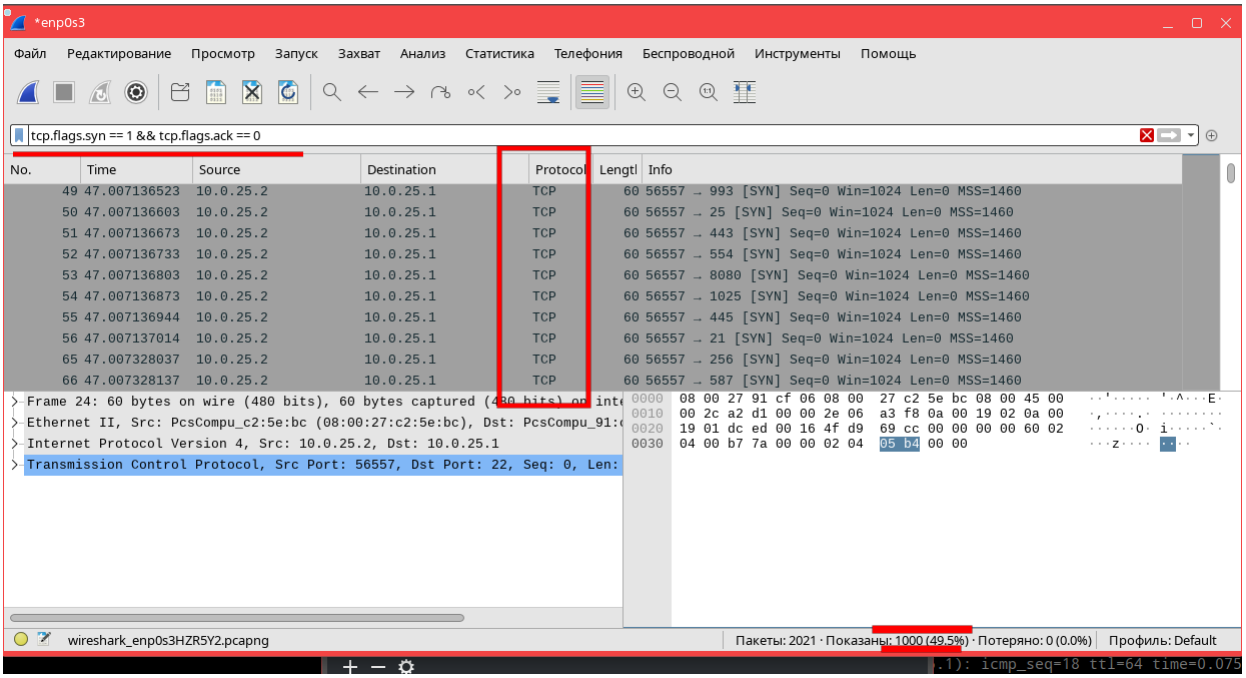

Рисунок 14 – вывод Wireshark с применённым фильтром

Данный эксперимент демонстрирует, как с помощью Wireshark и его фильтров можно эффективно обнаруживать подозрительную активность в сети, такую как сканирование портов. Это важно для своевременного выявления потенциальных угроз и принятия мер по защите сетевой инфраструктуры. Используя фильтры для выделения специфических типов трафика (например, SYN-пакетов), администраторы сети могут оперативно реагировать на попытки несанкционированного доступа и минимизировать риски безопасности.

Вывод

В ходе лабораторной работы было успешно освоено использование сетевого анализатора Wireshark для анализа трафика сети. Применение различных фильтров позволило эффективно отслеживать и идентифицировать специфические типы сетевого трафика, такие как SYN-пакеты, что является важным аспектом в обнаружении сканирования портов. Эксперимент продемонстрировал, как можно выявлять подозрительную активность в сети, своевременно реагировать на потенциальные угрозы и предотвращать несанкционированный доступ. Установка и настройка необходимых инструментов, таких как Nmap, показали важность интеграции различных программ для комплексного анализа безопасности сети. Практическое применение фильтров Wireshark облегчает процесс диагностики и позволяет администраторам сети сосредоточиться на значимых данных, исключая ненужный трафик. Кроме того, работа подчеркнула значение знания различных сетевых протоколов и их поведения для эффективного мониторинга и защиты сетевой инфраструктуры. В целом, лабораторная работа способствовала углублению понимания принципов сетевой безопасности и укреплению навыков использования профессиональных инструментов для обеспечения надежности и безопасности информационных систем.

Контрольные вопросы

1. Для чего используется wireshark?

Wireshark – это программный анализатор протоколов. Он применяется для захвата и анализа сетевого трафика на низком уровне, позволяя исследовать пакеты данных, смотреть их заголовки и содержимое, определять протоколы, диагностировать проблемы с сетью, находить уязвимости и аномалии в трафике.

2. Что такое фильтры?

Фильтры – это специальные выражения, позволяющие отбирать из всего захваченного сетевого трафика только те пакеты, которые соответствуют заданным критериям. Фильтры упрощают анализ, позволяя быстро исключить лишний «шум» и сосредоточиться на нужных данных. В Wireshark существуют фильтры захвата (capture filters) и фильтры отображения (display filters).

3. Какие протоколы существуют?

Сетевых протоколов очень много, они работают на разных уровнях модели OSI/TCP-IP. Примеры:

Канальные протоколы: Ethernet, PPP

Сетевые протоколы: IPv4, IPv6, ARP

Транспортные протоколы: TCP, UDP

Прикладные протоколы: HTTP, HTTPS, FTP, DNS, SMTP, NTP, SSH и многие другие.