**Лабораторна робота № 4 з предмету «Комп’ютерні інформаційні мережі»**

Виконав:

студент групи ПМІ-32

Юрас Назар

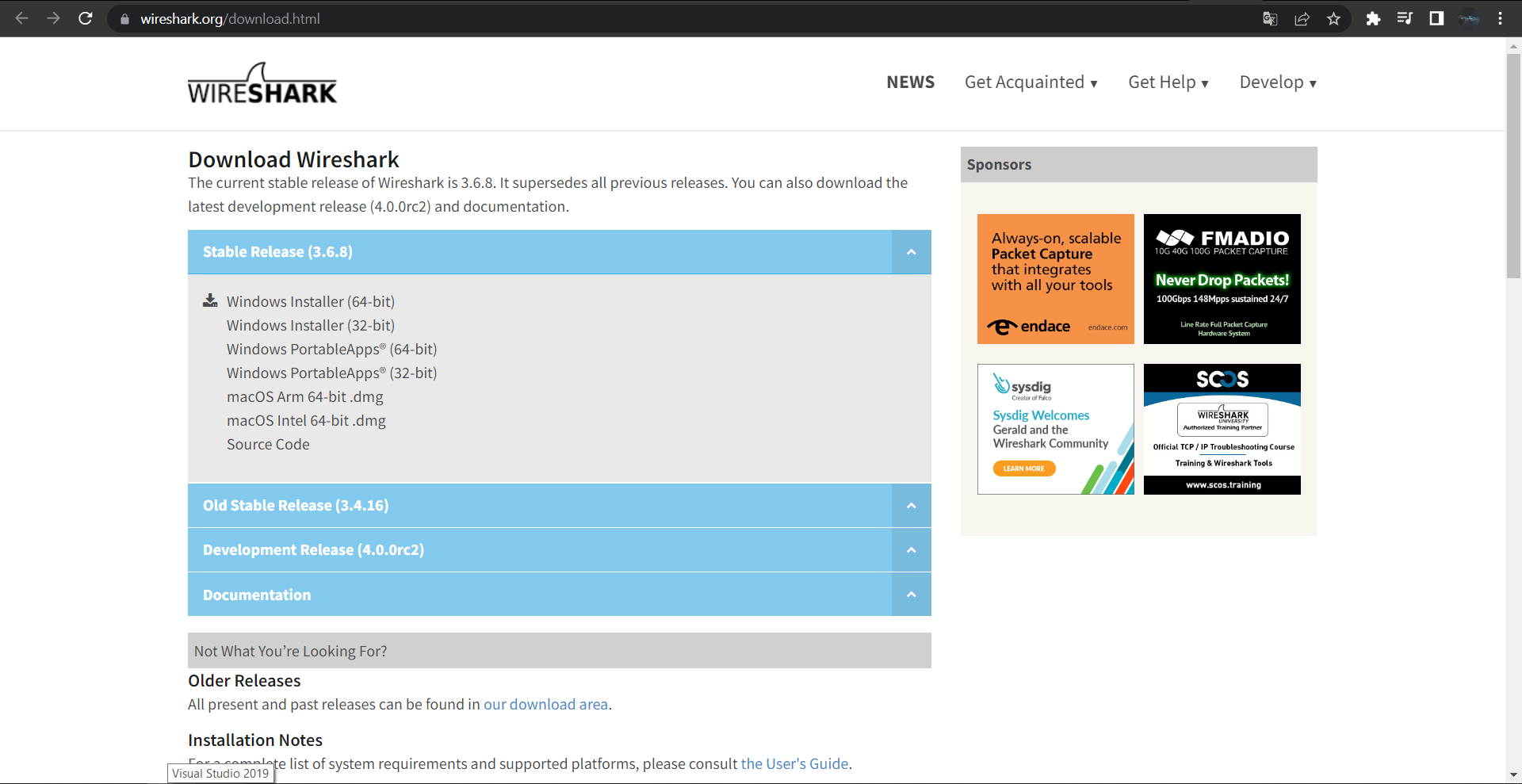
Львів-2022

**Тема:** інтерфейс аналізатора пакетів Wireshark

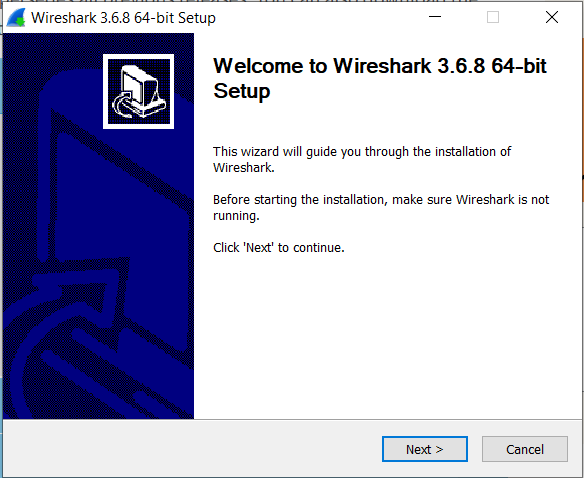
**Мета:** Отримати загальні уявлення про функціональні можливості аналізатора мережевих пакетів Wireshark, ознайомитися з графічним інтерфейсом програми, навчитися захоплювати, сортувати та фільтрувати пакети.

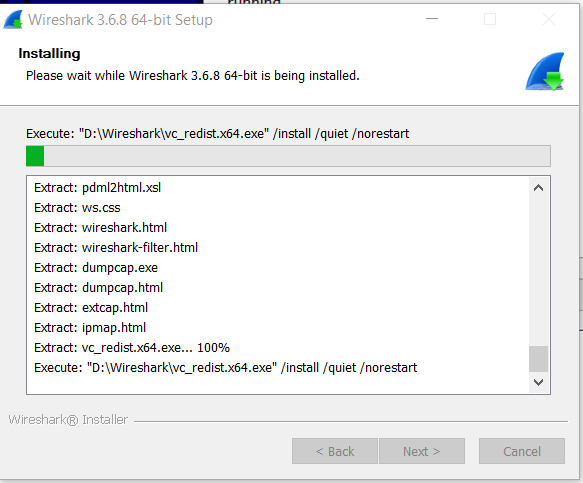
**Хід роботи:**

Зайшов на офіційний сайт Wireshark та завантажив програму

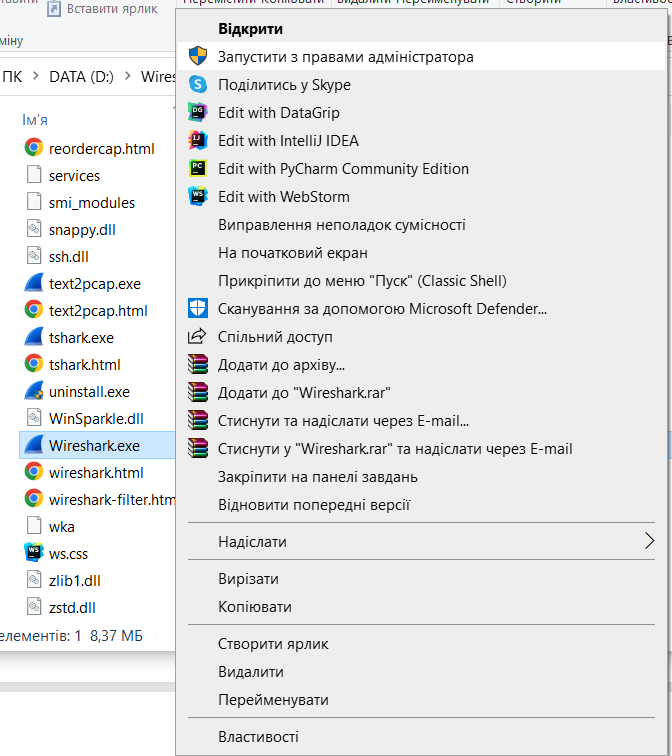


Починаємо встановлення:

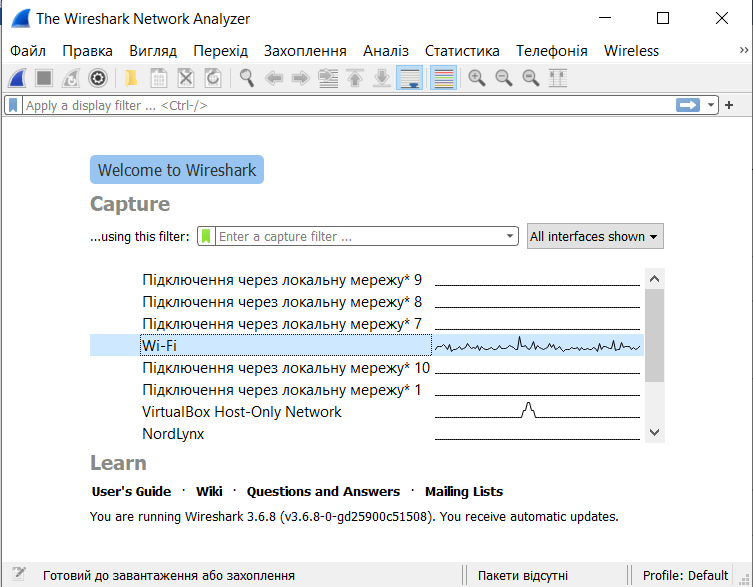




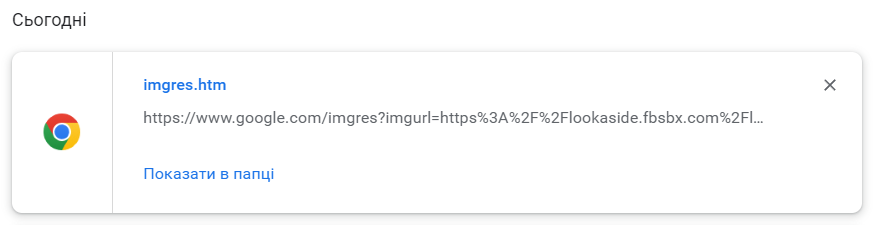
Далі запускаю WireShark від імені адміністратора:



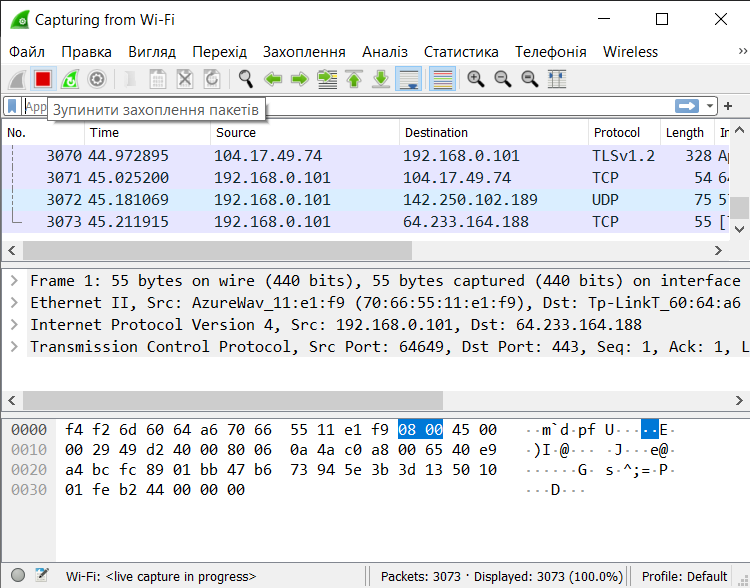
Вибираю потрібну мені мережу (в моєму випадку це Wi-Fi) і натискаю кнопку «Почати захоплення пакетів»)



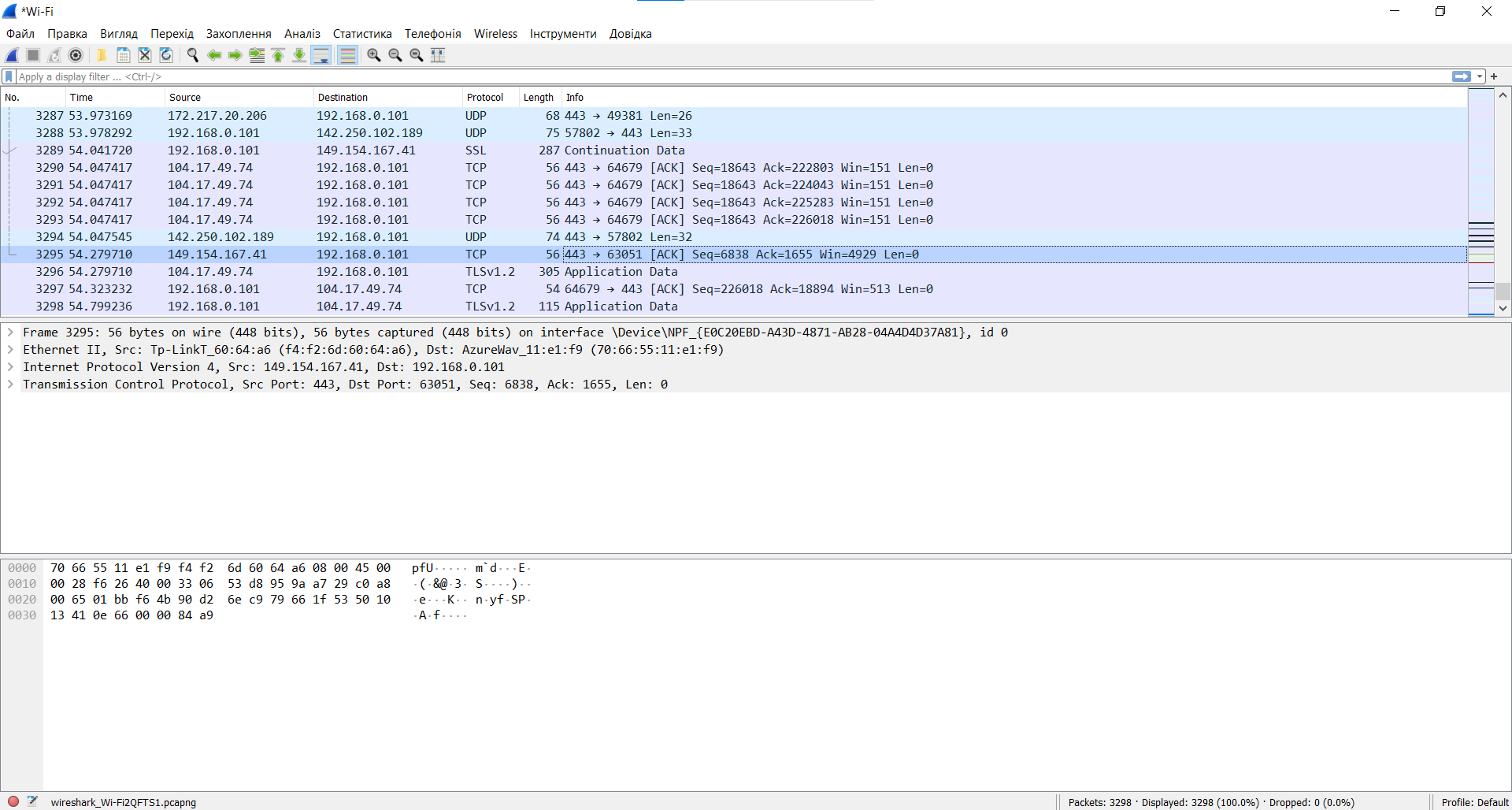
Для створення якоїсь активності у браузері я завантажив зображення:



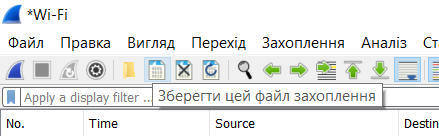
Далі зупиняю перехоплення пакетів:



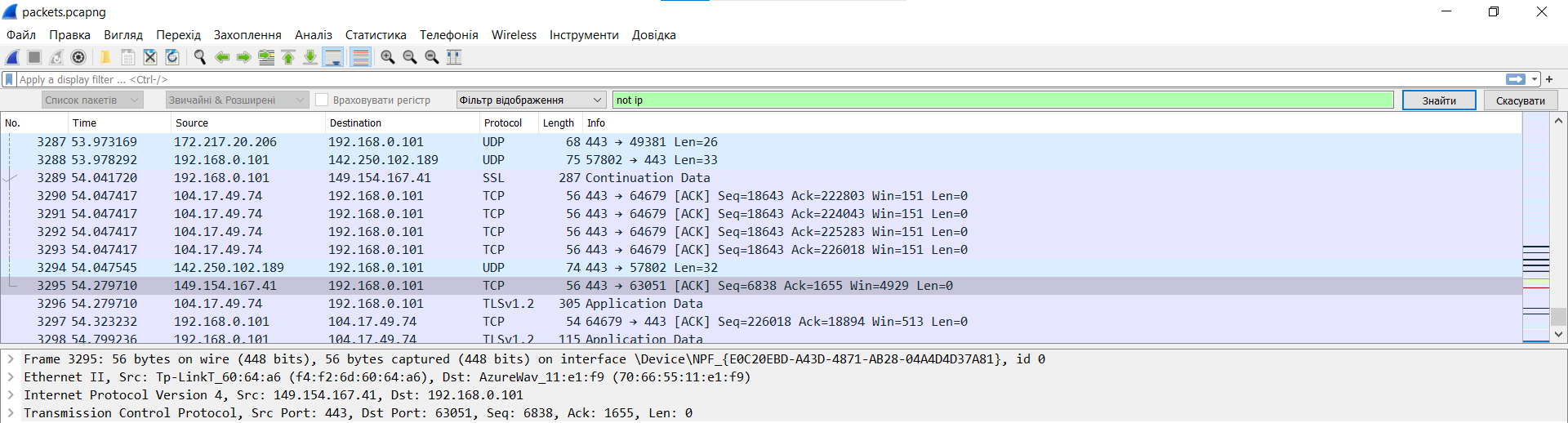
Ознайомився з трьома основними елементами вікна:



Зберігаю захоплені пакети у файл:

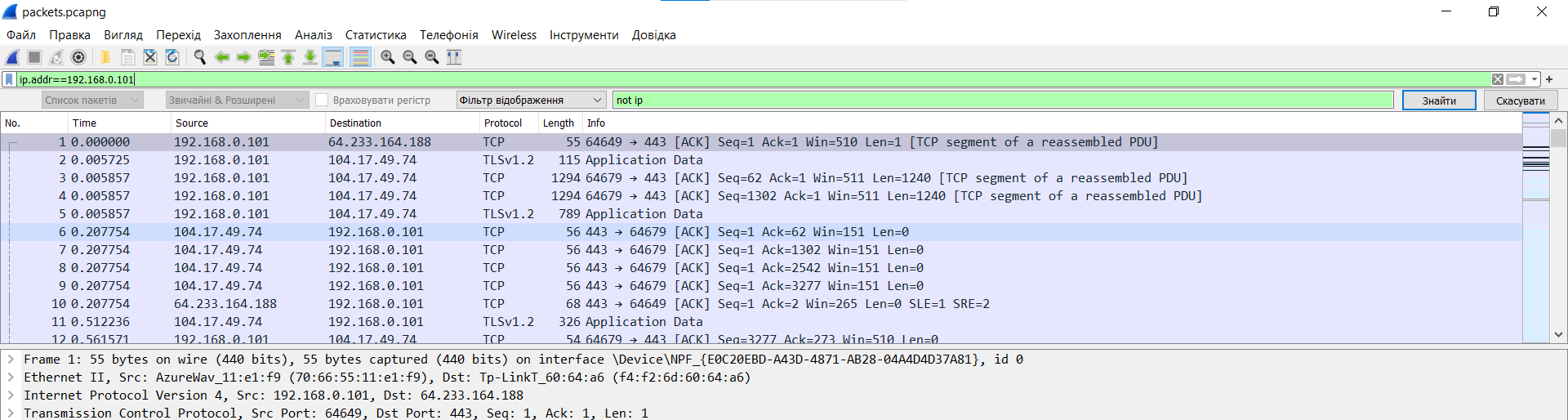


Далі за допомогою пошуку шукаю пакети, які не стосуються протоколу IP, використовуючи вираз **not ip**:

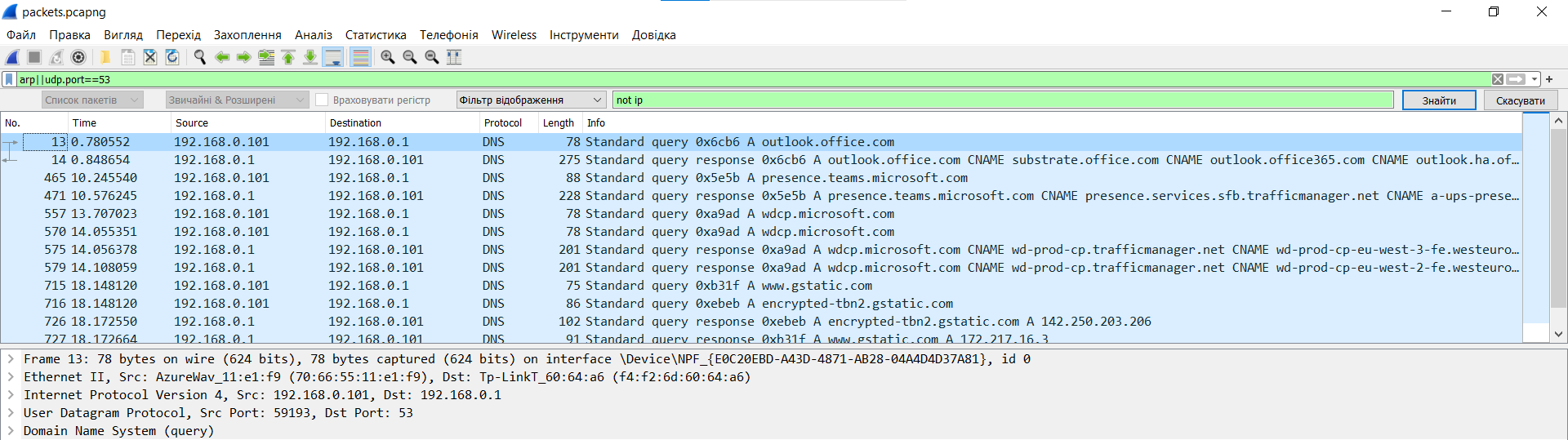


Поекспериментував з деякими виразами:

ip.addr==192.168.0.101, пошук по конкретній ір-адресі:

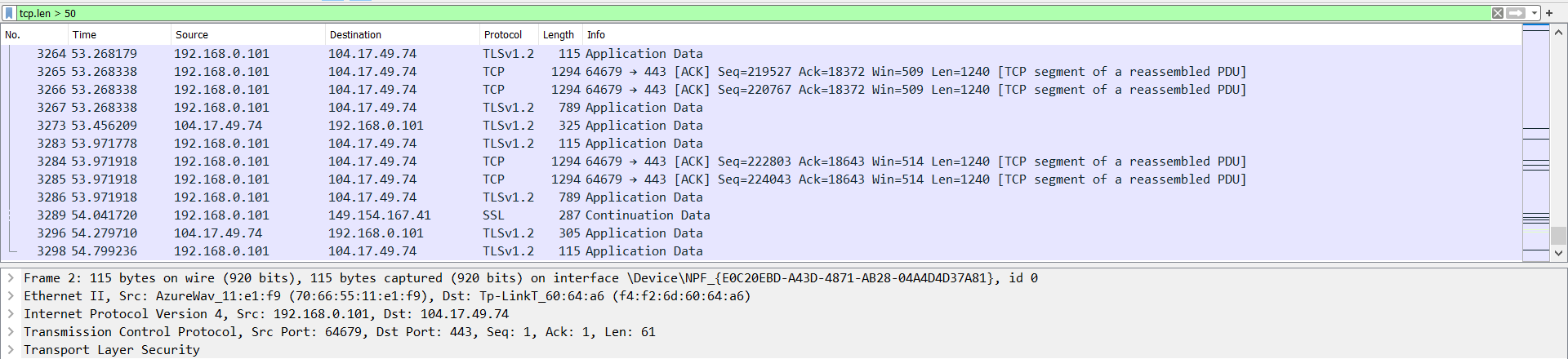


arp||udp.port==53 – це фільтр по протоколу, в даному випадку по arp або udp, показує вихідний трафік який направляється за межі локальної мережі. Доволі хороший спосіб визначити ПЗ яке взаємодіє з інтернетом по незвичних протоколах:

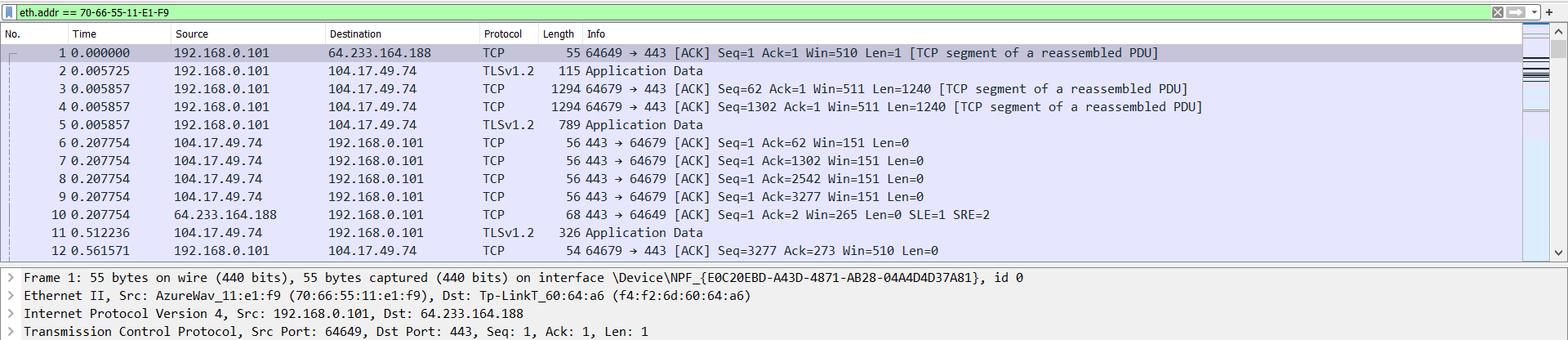


Команда для пошуку по кількості байт:

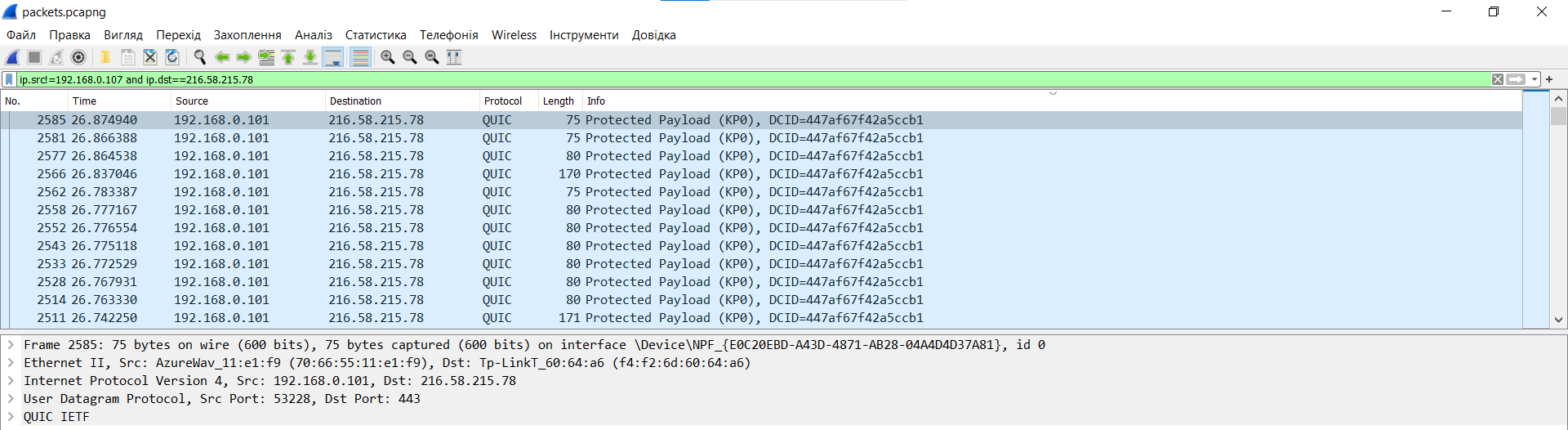
tcp.len > 50



Команда для пошуку пакетів за Ethernet-протоколом за конкретним значенням: eth.addr== 70-66-55-11-E1-F9.

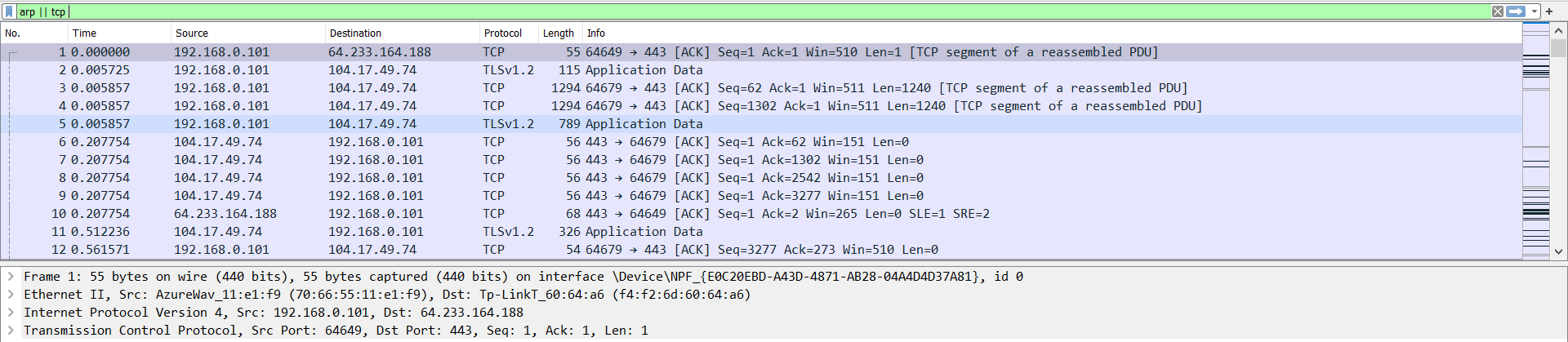


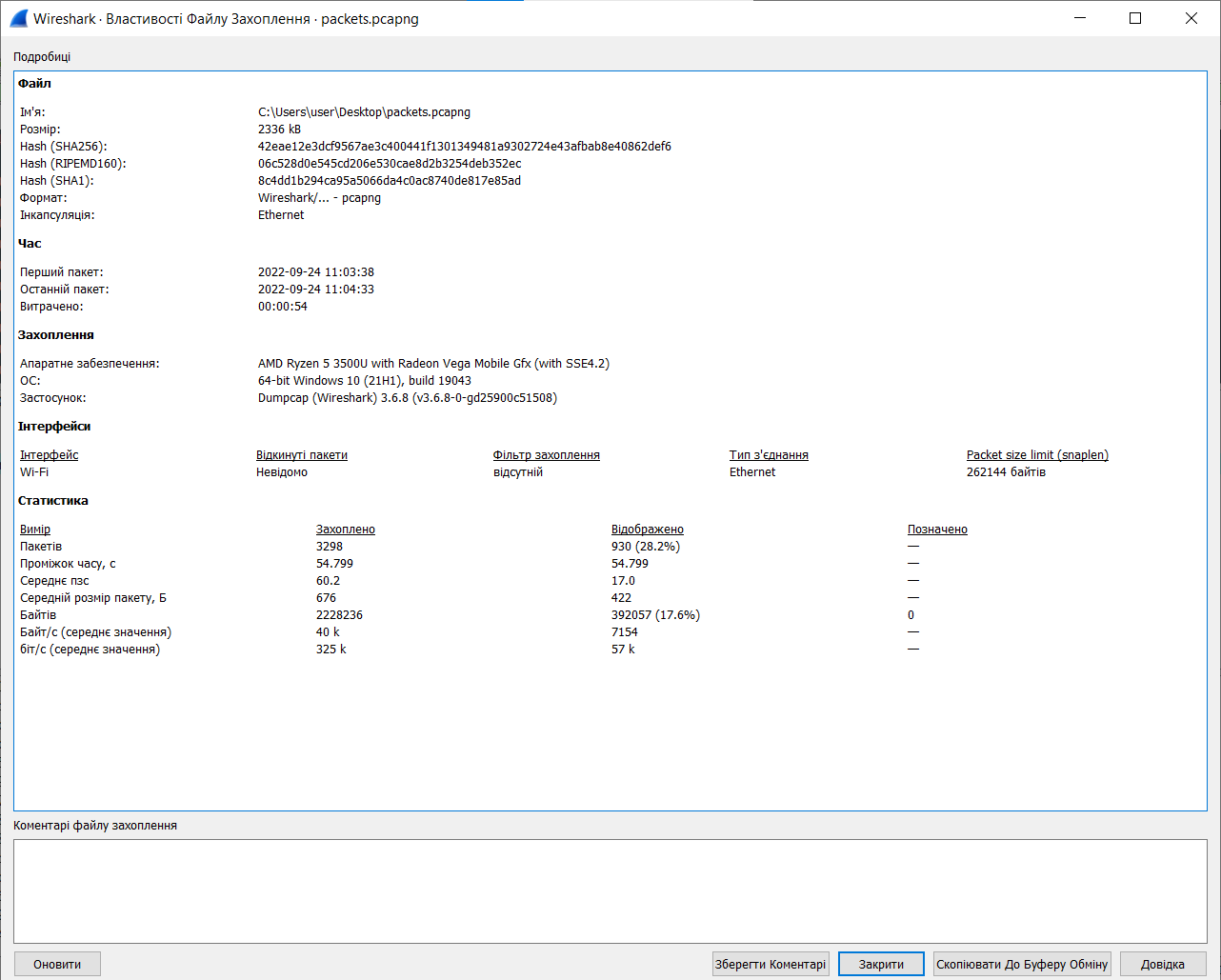
Команди, які показують весь трафік у локальній мережі окрім заданого значення, між робочими станціями та без Інтернету: ip.src!=192.168.0.107 ip.dst==216.58.215.78



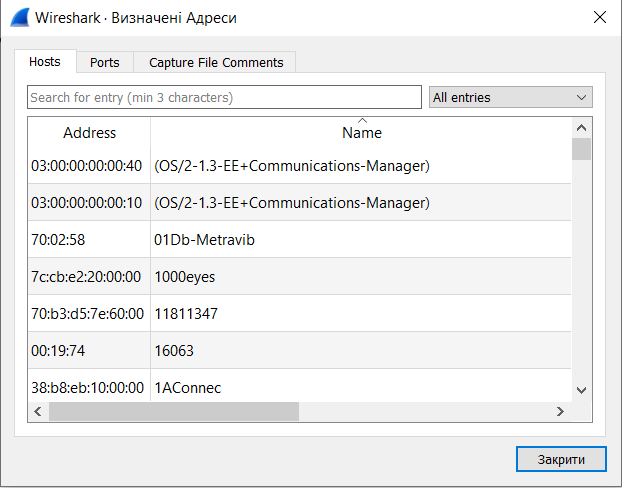
Ну і наостанок фільтр по протоколах: http&&ftp&&arp, показує всі пакети де протоколи дорівнюють http, ftp та arp. В моєму випадку таких не виявилось, тому я використав ось такий фільтр:

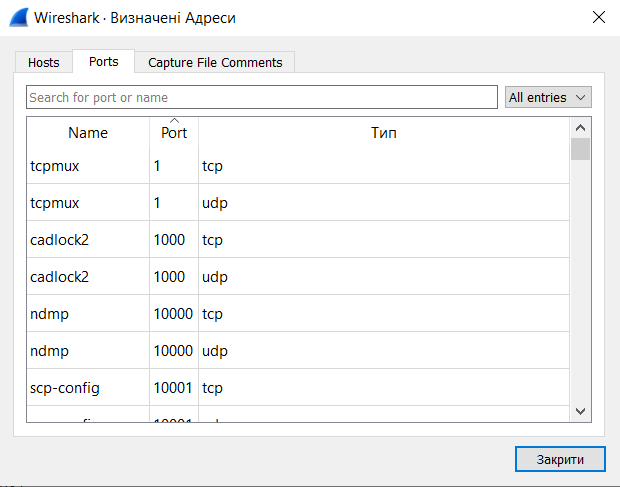
arp || tcp.



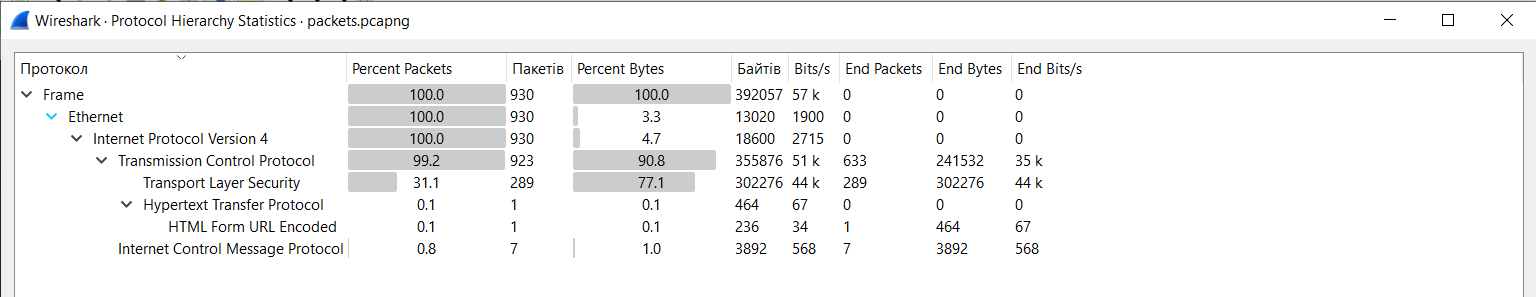
Далі у вкладці «Статистика» ознайомився з такими пунктами як Capture File Properties. В ньому описані загальні характеристики файлу захоплення, по типу імені, розміру, часу захоплення першого пакету, інтерфейсів і так далі

Resolved addresses: в цьому вікні вказані адреси та ім’я хостів, інформація про порти та коментарі файлу

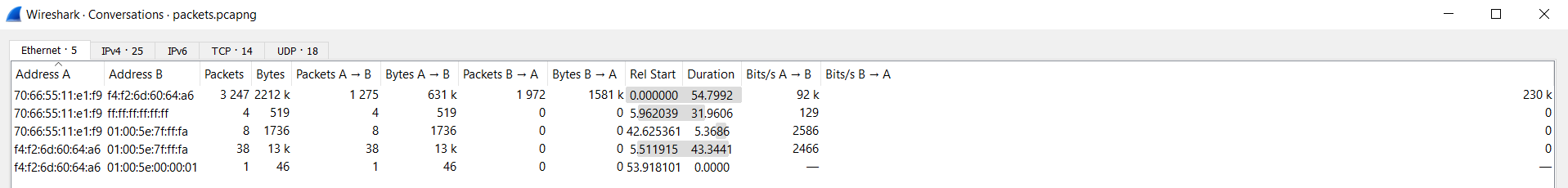




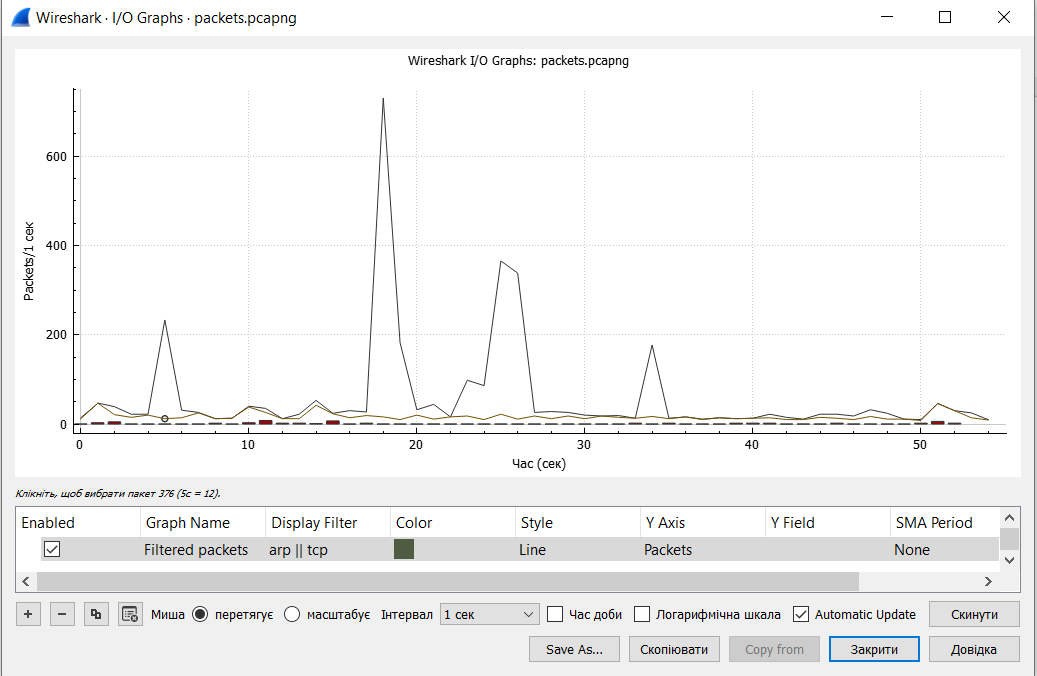
Protocol Hierarchy: ну тут все просто, ієрархія протоколів файлу



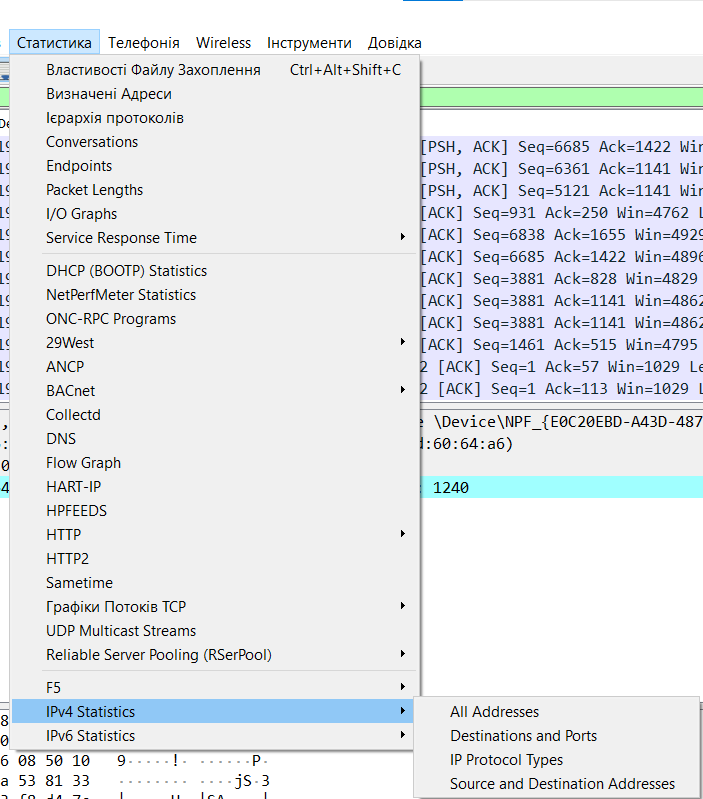
Conversations: повна інформація про передавання пакетів між двома адресами, типу кількості байт, кількості пакетів, тощо



I/O Graphs: графік, який показує кількість пакетів, переданих за секунду



IPv4 Statistics: показує всі ІР-адреси, шляхи призначення та порти, типи ІР-протоколів, джерела та шляхи призначення



Висновок: на цій лабораторній роботі я отримав загальні уявлення про функціональні можливості аналізатора мережевих пакетів Wireshark, ознайомився з графічним інтерфейсом програми, навчився захоплювати, сортувати та фільтрувати пакети.