Wireshark Basics

Network Packet Analysis Fundamentals

# Overview

**Wireshark** is an **open-source, cross-platform network packet analyser tool** capable of sniffing and investigating live traffic and inspecting packet captures (PCAP). It is commonly used as one of the best packet analysis tools in the industry.

# Learning Objectives

* **Navigate and configure Wireshark** – Learn the interface and essential settings
* **Inspect packets and discover information** – Analyze different layers of TCP/IP protocol stack
* **Apply display filters** – Filter and focus on specific network traffic

# Lab Files Reference

The following capture files are provided for hands-on practice:

|  |  |
| --- | --- |
| **File Name** | **Purpose** |
| http1.pcapng | Simulate actions shown in screenshots (demo file) |
| Exercise.pcapng | Answer the questions (exercise file) |

# Use Cases

Wireshark is one of the most potent traffic analyser tools available. Key purposes include:

* **Network Troubleshooting** – Detecting network problems such as load failure points and congestion
* **Security Analysis** – Detecting security anomalies like rogue hosts, abnormal port usage, and suspicious traffic
* **Protocol Investigation** – Learning protocol details such as response codes and payload data

**⚠️ Important:** Wireshark is NOT an Intrusion Detection System (IDS). It only allows analysts to discover and investigate packets. It does not modify packets; it reads them. Detecting anomalies relies on the analyst's knowledge and skills.

# GUI and Data

Wireshark GUI opens with a single all-in-one page with five main sections:

|  |  |
| --- | --- |
| **Section** | **Description** |
| Toolbar | Main toolbar with menus and shortcuts for packet sniffing, filtering, sorting, summarising, exporting and merging |
| Display Filter Bar | Main query and filtering section |
| Recent Files | List of recently investigated files (double-click to recall) |
| Capture Filter & Interfaces | Capture filters and available sniffing points (network interfaces like lo, eth0, ens33) |
| Status Bar | Tool status, profile and numeric packet information |

# Loading PCAP Files

**Methods to load PCAP files:** Use the "File" menu, drag and drop the file, or double-click on the file.

Once loaded, packet details are shown in three panes:

|  |  |
| --- | --- |
| **Pane** | **Description** |
| Packet List Pane | Summary of each packet (source/destination addresses, protocol, packet info). Click to select for investigation. |
| Packet Details Pane | Detailed protocol breakdown of the selected packet |
| Packet Bytes Pane | Hex and decoded ASCII representation of the selected packet |

# Colouring Packets

Wireshark colours packets based on different conditions and protocols to help spot anomalies quickly.

**Two types of colouring methods:**

* **Temporary Rules** – Only available during program session. Access via right-click menu or "View → Conversation Filter"
* **Permanent Rules** – Saved under preference file (profile). Access via right-click menu or "View → Coloring Rules"

💡 Tip: Use "Colourise Packet List" menu to activate/deactivate colouring rules.

# Traffic Sniffing

Control buttons for traffic capture:

|  |  |
| --- | --- |
| **Button** | **Action** |
| 🦈 Blue Shark Button | Start network sniffing (capture traffic) |
| 🔴 Red Button | Stop sniffing |
| 🟢 Green Button | Restart sniffing process |

The status bar shows the sniffing interface and number of collected packets.

# Merge PCAP Files

**Path:** File → Merge

1. Select second file to merge with the currently processed one
2. Wireshark shows total packet count in selected file
3. Click "Open" to merge files into a new pcap
4. Save the merged file before working on it

# View File Details

**Path:** Statistics → Capture File Properties (or click pcap icon at bottom left)

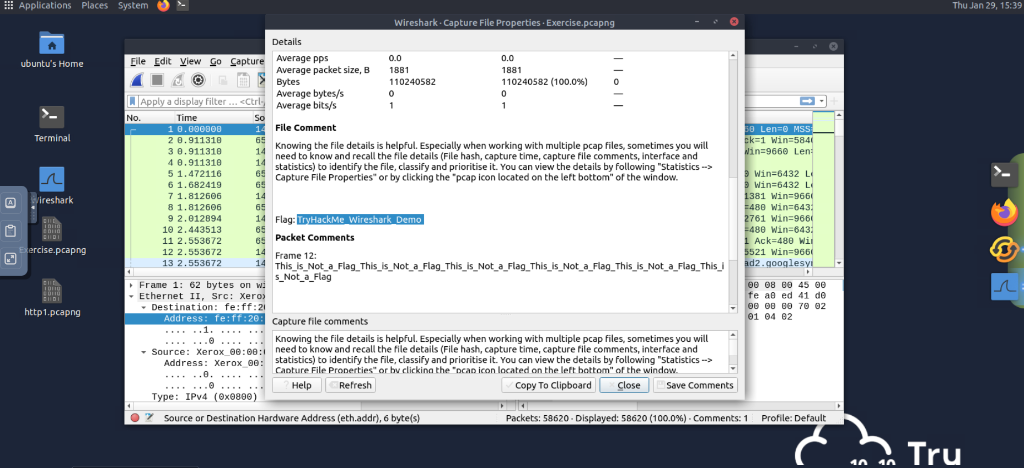
File details include:

* File hash
* Capture time
* Capture file comments
* Interface information
* Statistics

# Task 2: Tool Overview

Answers from the Capture File Properties analysis:

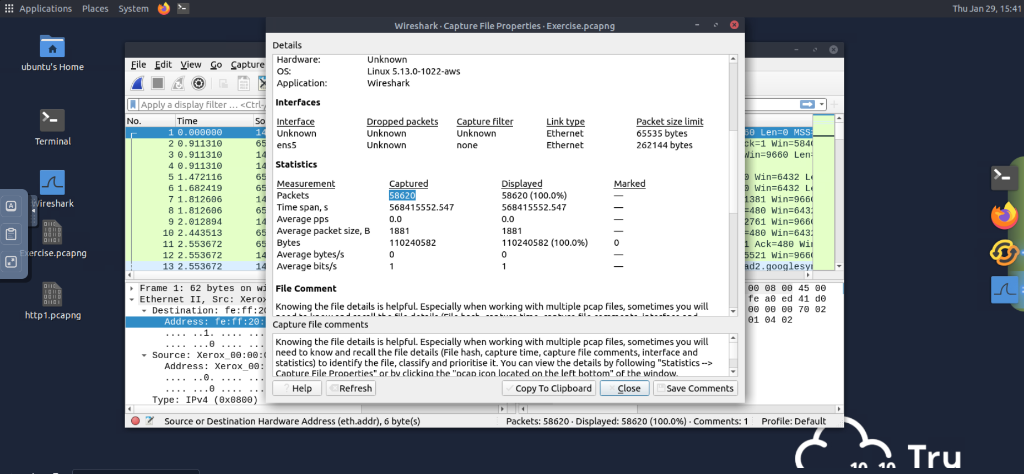
**Q1: What is the flag in the File Comment?**



*Screenshot: File Comment showing flag "TryHackMe Wireshark Demo"*

**Answer:** TryHackMe Wireshark Demo

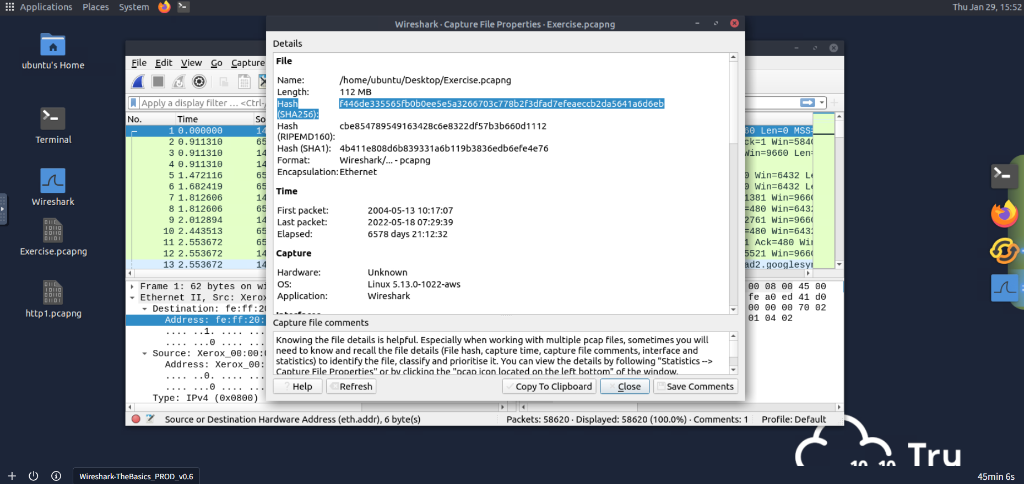
**Q2: What is the total number of packets?**



*Screenshot: Statistics showing total packets captured*

**Answer:** 58620

**Q3: What is the SHA256 hash value?**



*Screenshot: File Hash details*

**Answer:** f446de335565fb0b0ee5e5a3266703c778b2f3dfad7efeaeccb2da5641a6d6eb

# Task 3: Packet Dissection

Packet dissection (protocol dissection) investigates packet details by decoding protocols and fields. Wireshark breaks down packets into 5-7 layers based on the OSI model:

* **Frame (Layer 1)** – Physical layer details
* **Source [MAC] (Layer 2)** – Data Link layer (Source/Dest MAC)
* **Source [IP] (Layer 3)** – Network layer (Source/Dest IPv4)
* **Protocol (Layer 4)** – Transport layer (UDP/TCP, ports)
* **Application Protocol (Layer 5)** – Application layer (HTTP, FTP, SMB)
* **Application Data** – Application-specific data

The following exercises use packet number 38 from "Exercise.pcapng".

**Q1: Which markup language is used under the HTTP protocol?**



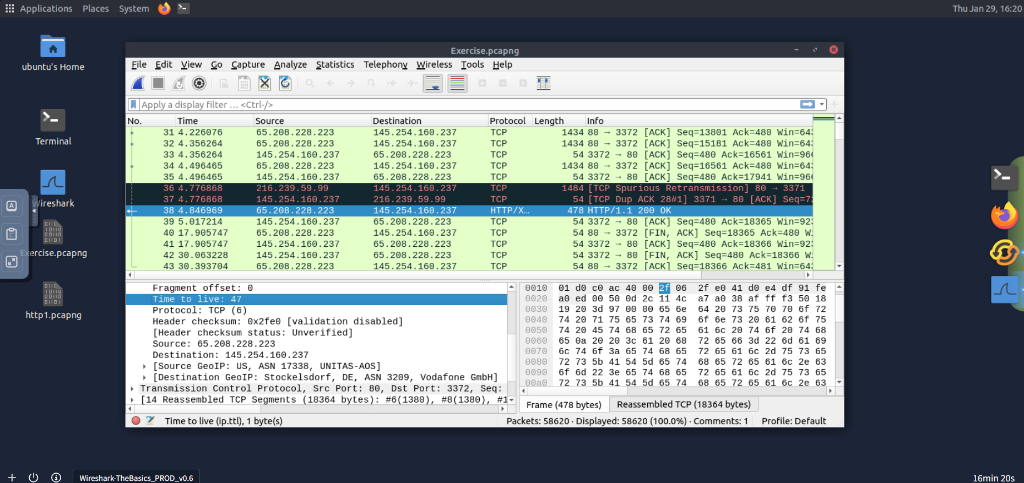
Answer: eXtensible Markup Language

**Q2: What is the arrival date of the packet? (Month/Day/Year)**



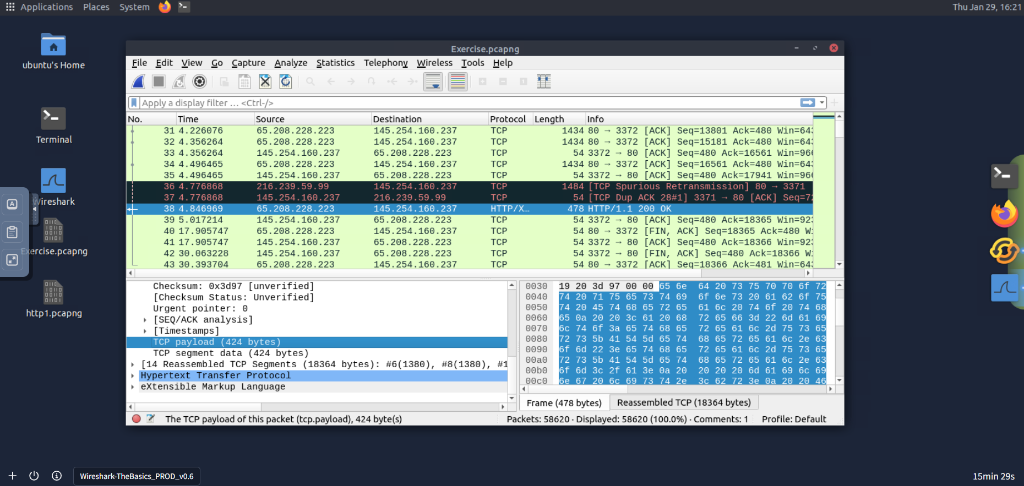
Answer: 05/13/2004

**Q3: What is the TTL value?**



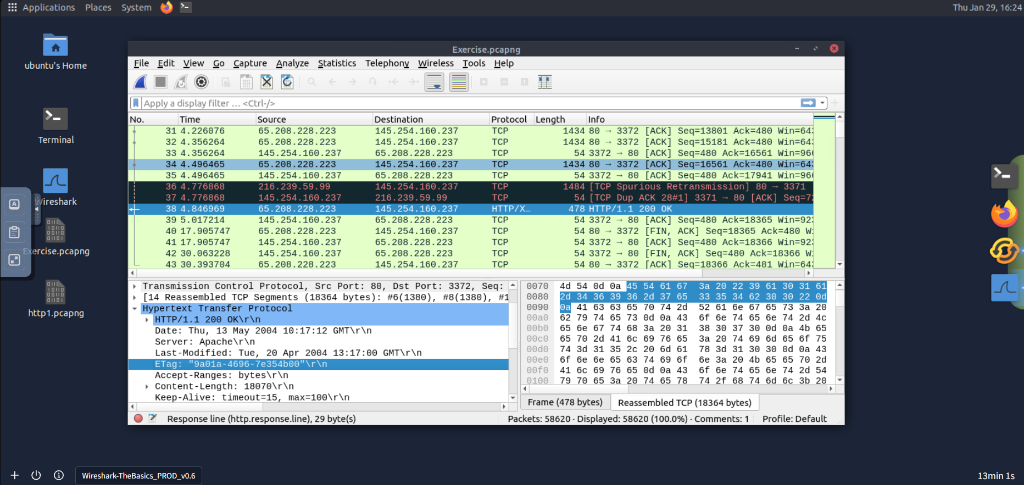
Answer: 47

**Q4: What is the TCP payload size?**



Answer: 424

**Q5: What is the e-tag value?**



Answer: 9a01a-4696-7e354b00

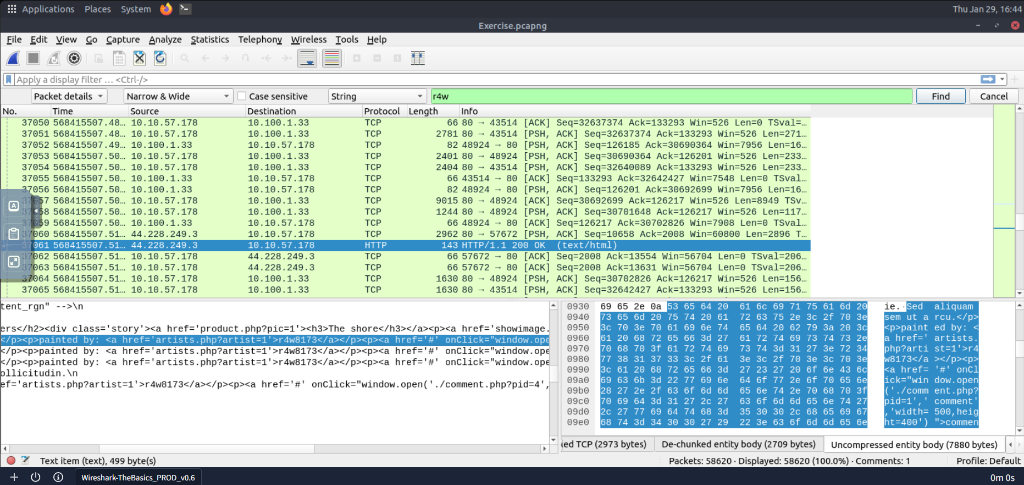
# Task 4: Packet Navigation & Investigation

This section covers essential navigation and investigation tools in Wireshark:

|  |  |
| --- | --- |
| **Feature** | **Description** |
| Packet Numbers | Unique ID assigned to each packet for indexing and easy reference. |
| Go to Packet | Navigation tool to jump to specific packets or track conversations. |
| Find Packets | Search for packets by content using Display Filter, Hex, String, or Regex. Note: Select the correct pane (List, Details, Bytes) for the search. |
| Mark Packets | Highlight packets in black for further investigation. Markers are session-specific. |
| Packet Comments | Add persistent notes to packets for collaboration or later analysis. |
| Export Packets | Save specific packets (e.g., suspicious ones) to a separate file. |
| Export Objects | Extract files (HTTP, SMB, etc.) transferred over the wire. |
| Time Display Format | Change timestamp format (e.g., to UTC) via "View → Time Display Format". |
| Expert Info | Automatic anomaly detection with severity levels: Chat (Blue), Note (Cyan), Warn (Yellow), Error (Red). |

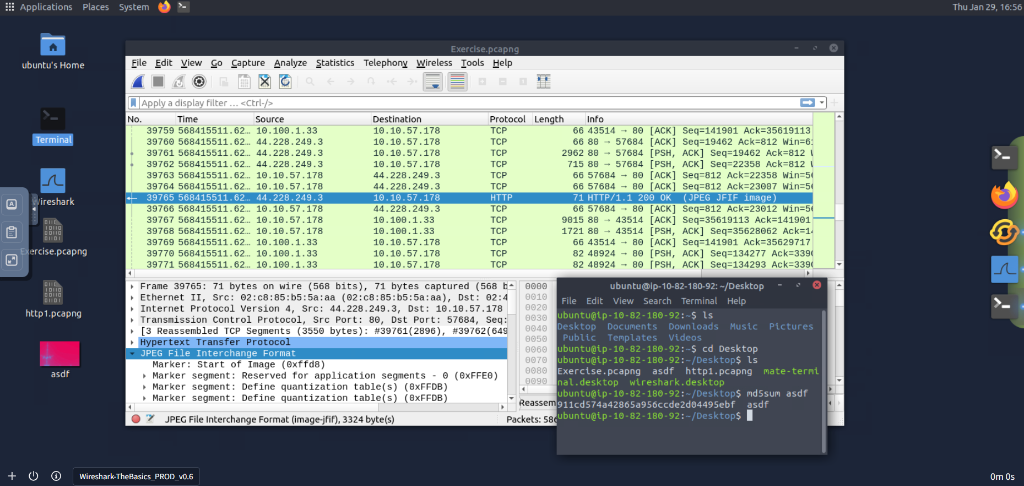
The following exercises use "Exercise.pcapng".

**Q1: Search for "r4w" string in packet details. What is the name of artist 1?**



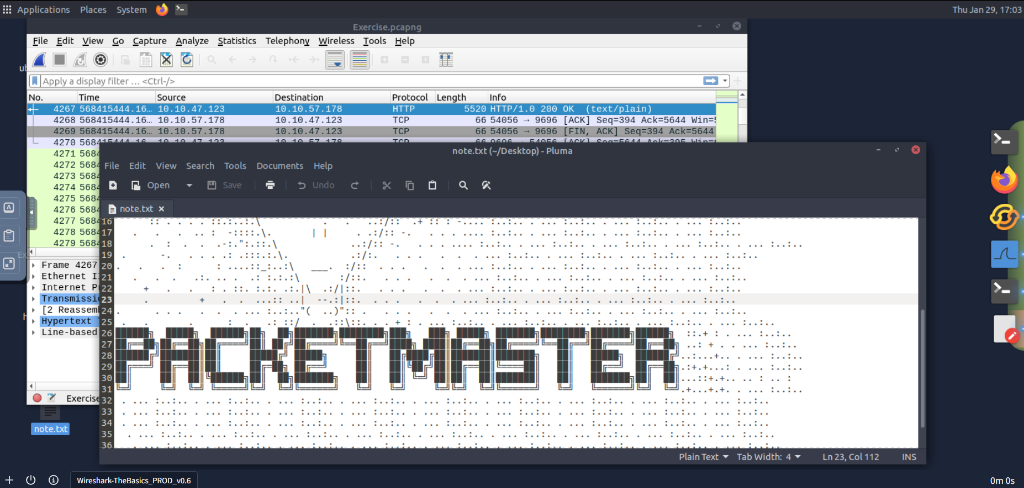
Answer: r4w8173

**Q2: Go to packet 12 and read the packet comments. What is the answer?**



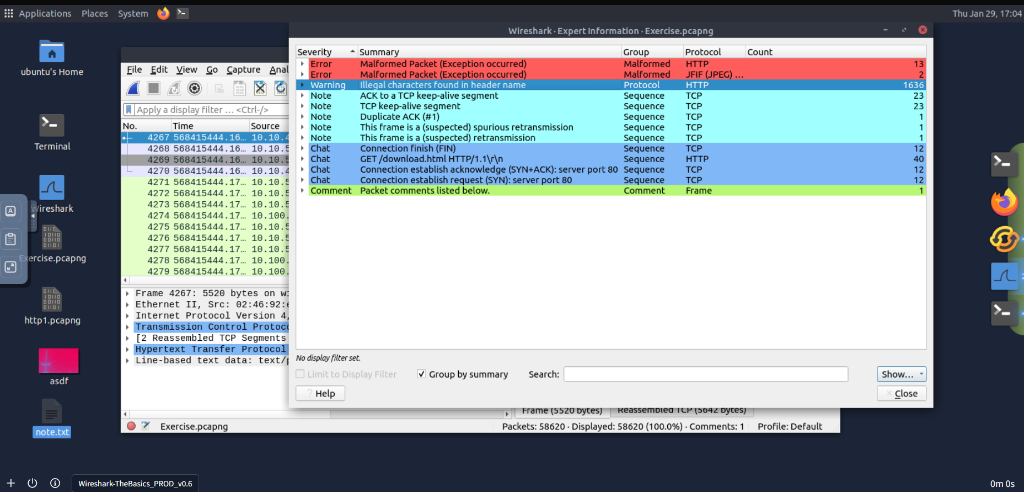
Answer: 911cd574a42865a956ccde2d04495ebf

**Q3: There is a ".txt" file inside the capture file. Find the file and read it; what is the alien's name?**



Answer: PACKETMASTER

**Q4: Look at the expert info section. What is the number of warnings?**



Answer: 1636

# Task 5: Packet Filtering

Wireshark provides powerful filtering capabilities to isolate specific traffic. The two main types are Capture filters (pre-capture) and Display filters (post-capture). Key filtering methods include:

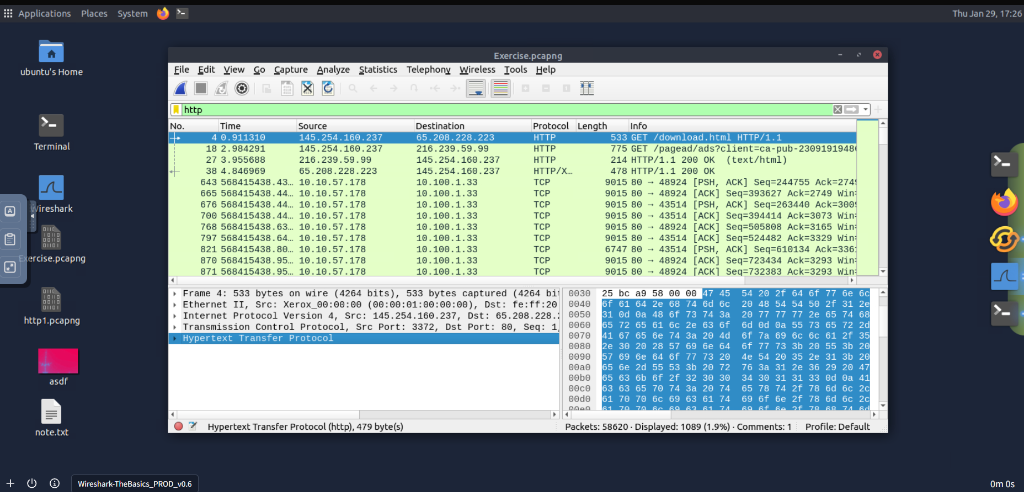
|  |  |
| --- | --- |
| **Method** | **Description** |
| Apply as Filter | Right-click a value to quickly filter for it. |
| Conversation Filter | Focus on a specific conversation (IPs/Ports) and hide others. |
| Colourise Conversation | Highlight specific conversations without hiding other packets. |
| Prepare as Filter | Adds the filter query to the bar without executing it immediately (allows combining queries). |
| Apply as Column | Add a specific packet field as a column in the Packet List pane. |
| Follow Stream | Reconstruct application-level data to view the full conversation (TCP/UDP/HTTP) in a readable format. |

Common Simple Filters:

* Protocol: http, tcp, udp, dns
* Port: tcp.port == 80
* IP Address: ip.addr == 192.168.1.1

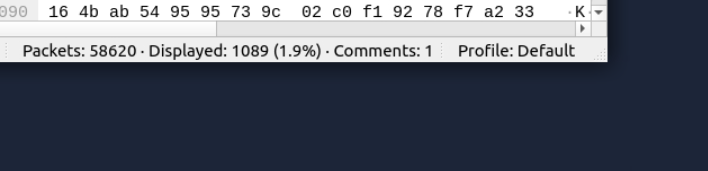
The following exercises use "Exercise.pcapng".

**Q1: Go to packet 4. Right-click "Hypertext Transfer Protocol" -> Apply as Filter. What is the filter query?**



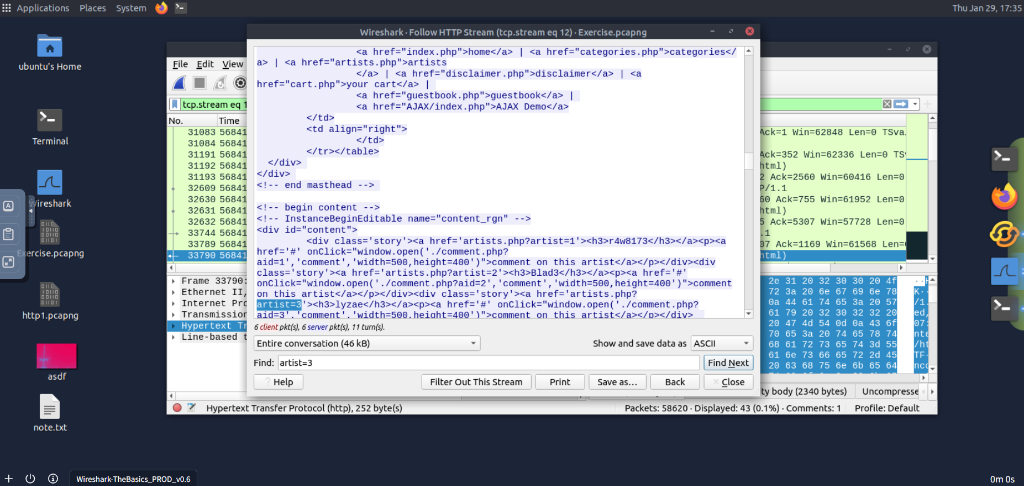
Answer: http

**Q2: What is the number of displayed packets?**



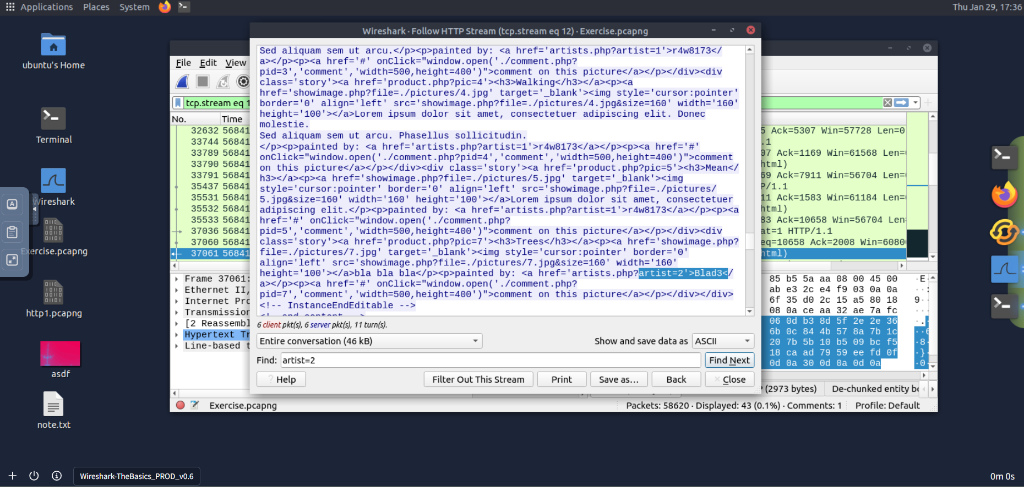
Answer: 1089

**Q3: Go to packet 33790, follow stream. What is the total number of artists?**



Answer: 3

**Q4: What is the name of the second artist?**



Answer: Blad3

# Quick Reference: Q&A

**Q:** Which file is used to simulate the screenshots?

**A:** http1.pcapng ✓

**Q:** Which file is used to answer the questions?

**A:** Exercise.pcapng ✓