Timestamp,Src\_IP,Dest\_IP,Protocol,Source\_Port,Destination\_Port,Packet\_Size,Status

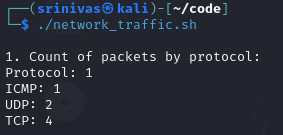
2024-09-30 10:15:10,192.168.1.10,172.217.12.206,TCP,443,51413,1500,Accepted 2024-09-30 10:15:12,192.168.1.15,203.0.113.5,UDP,53,55432,512,Dropped 2024-09-30 10:15:14,10.0.0.2,192.168.1.10,TCP,80,61324,1420,Accepted 2024-09-30 10:15:16,172.16.0.5,192.168.1.15,ICMP,64,Dropped 2024-09-30 10:15:18,192.168.1.10,198.51.100.23,TCP,443,1025,1500,Accepted 2024-09-30 10:15:20,198.51.100.23,192.168.1.10,TCP,443,1025,1400,Accepted 2024-09-30 10:15:22,203.0.113.5,192.168.1.15,UDP,123,49152,512,Dropped

Src IP – Source Ip

Dest Ip – Destination Ip

Write a awk script to

1. count how many packets use each protocol (TCP, UDP, ICMP) 



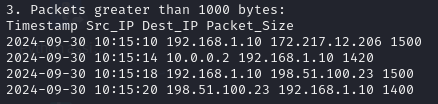
2. filter and print only the dropped packets.





3. print the Timestamp, Source\_IP, Destination\_IP, and Packet\_Size for packets that have a size greater than 1000 bytes.





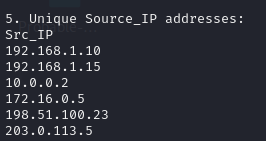
4. display traffic that is directed to destination port 443.





5. print all unique Source\_IP addresses from the network\_traffic.csv file.





6. filter only TCP traffic and calculate the average packet size.





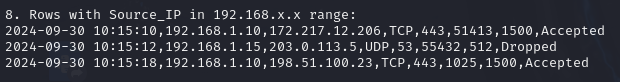
7. Count invalid records





8. extract and print all rows where the Source\_IP is in the 192.168.x.x range.





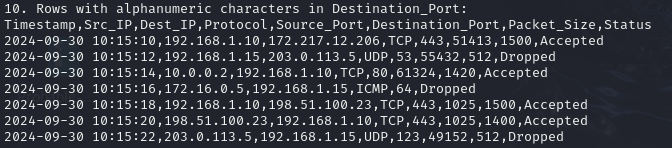
9. match traffic directed to either port 80 (HTTP) or port 443 (HTTPS).





10. filter out rows where the Destination\_Port contains any alphanumeric characters (letters or numbers).





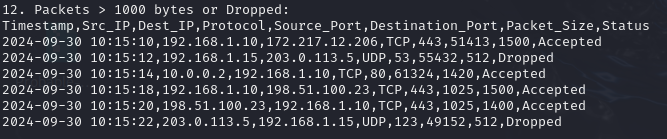
11. filter out traffic where the protocol is TCP AND the destination port is 443 (HTTPS traffic).





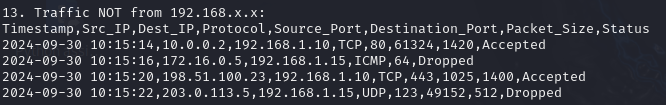
12. filter out and print traffic where the Packet\_Size is greater than 1000 OR the Status is Dropped.





13. print traffic NOT originating from 192.168.x.x IP addresses.





14. filter rows where both Source\_IP and Destination\_IP are within the 192.168.x.x range.





15. filter out traffic where the destination port is 22 OR the packet size is less than 100 bytes.



