Assignment

by 2022512 .

Submission date: 12-Dec-2024 07:23AM (UTC-0800)
Submission ID: 2550245136
File name: 2022512_NS_PROJECT.pdf (942.62K)
Word count: 3554
Character count: 19340

Network Security & Information Assurance (CY-331)

Project Report

Network Traffic Virtualization Using Wireshark and Google Maps

Group Members

Saad Ali (2022512)
Ahmed Rashid (2022677)
Aurangzaib Bhatti (2022355)

Submission Date: 11/12/2024

Professor: Dr. Zain Siddique

Contents

Abstract

It is important for security and to prevent threats in networks to recognize malicious traffic and properly define the geolocation of IP addresses in flow traffic. In this report, the author presents a system that was developed and implemented with the aims of possibly analyzing traffic on different networks with the help of Wireshark for packet capturing and GeoIP for geolocation services and Python for data processing as well as creating and developing a visualization system. Packet capture can be done from live traffic or packet capture files used by Wireshark that lets the extraction of source and destination Ip addresses. These IP addresses are then sorted using IP address ranges as public IP, private IP and multicasting IP. For public IPs location information like latitude and longitude is being obtained by GeoIP database which helps in getting the depth insight of traffic source and destination.

Consequently, the findings of this research demonstrate that the system is capable of capturing the Network Traffic and a possible malicious flow(s). The use of KML files made it easier to explore traffic flows as the bright lines connecting to areas of interest in California, where many technological firms are located infomed useful information as seen on Fig 5 below. The results of the geolocation analysis proved its applicability to network forensics; revealing overlooked traffic anomalies and helping to predict and prevent the occurrence of threats. The design of this system is also easily expandable for future addition of intrusion detection systems including SNORT making it possible to accurately detect security threats in real time.

The analysis provided showed both public and private traffic mix with much communication originating and going to identified servers in popular geographic domains. The geolocation analysis was extremely helpful to ID these acquaintances as having the significantly different traffic connectivity – such as from the different country or region – being possibly malicious, unauthorized or illegitimate connections. Private and multicast IPs, for example, 224.0.0.251 requires a previous set-up before it can be used in transmission.

1 Introduction

Upon the integration of networks in the contemporary world, it was essential to protect the network systems from the malicious activities. The proliferation of smart connectivity devices and the continued sophistication of threats mean that early identification and tracking of threats for response remain critical organizational practices. Traffic analysis is an effective tool used in heading and combating intruders with the aim of protecting networks against unauthorized access, and unlawful intrusion. From this review, it is clear that analyzing network traffic reveals threats, system performance issues that affect organizations and unauthorized access to valuable organizational information.

Today the most popular and at the same time powerful tool for packet capturing and analysis is Wireshark a network protocol analyzer. It enables them to monitor the traffic of packets which go through a particular network and volunteers a detailed view of data in the network. Due to the filtering, decoding, and analyzing single packets, Wireshark is a handy tool for network administrators and security analysts and researchers. It will also be helpful in finding out traffic anomalies, as well as in tracking down problems in networks, or even in conducting a forensic investigation.

The importance of this research is to explain how the combination of Wireshark, GeoIP, and Python work in analyzing network traffic. Thus, the presented study allows offering an integrated approach to the collection, analysis, and visualization of the network traffic data employing the mentioned tools. In particular, these findings show the usage scenario of this system in network protection: how to explore strange activities, how to discover unauthorized infiltration, and how to enhance the comprehension of a network environment. Packet capture and geolocation accompanied by the visualization of data provide a framework that could be easily extended to fit different objectives, ranging from post-incident analysis to real-time detection of threats.

Figure 1: Network Traffic Analysis

2 Related Work

Network traffic analysis remains a significant concern with a lot of research conducted on the utilities and approaches to use for the identification and prevention of illicit operations. This section discusses the prior studies in network traffic analyzing, intrusion detection systems and incorporating tools such as Wireshark, SNORT and GeoIP. The discussion is distilled into particular areas of focus that emphasise developments and issues within the domain.

2.1 Malicious Traffic Analysis Using Wireshark by the Collection of Indicators of Compromise

The next valuable element used to point to possible malicious activities in a network are known as Indicators of Compromise (IOCs). They are: source IP addresses, file hashes, domain name, and others that are usually linked to cyber threats. IOCs have been predominantly sourced from Wireshark, a packet capturing and analyzing program used in the extraction of information from traffic.

Earlier studies pointed out that Wireshark can dissect the packet at the network layer and give an idea of suspected patterns, including strange IP address, unfamiliar domain, or any ports that terminals should not be connected to. By applying filtering and analysis techniques, Wireshark can isolate packets associated with specific traffic patterns, enabling analysts to detect:

Malware traffic types of CC or Command and Control. Strange traffic activities for example traffic from multiple computers to a single external IP. Measures of Distributed Denial of Service (DDoS) attack types, such as SYN flood and malformed packets. As we know that Wireshark is user-friendly and quite flexible in capturing packets and also capable to dissect the traffic at multiple levels therefore it is selected as first choice of packet capturing tool for this study. The extracted IP addresses were used in further geolocation to provide more information about the source of the network traffic.

2.2 A Review Based on the Application of SNORT and Wireshark in Network Traffic Analysis

Even though Wireshark is essentially top of the line when it comes to traffic capture and analysis it does not support real-time intrusion detection. Another can be used in parallel with SNORT – also an open source Intrusion Detection System (IDS) whose function is in examining network packets online to search for suspicious patterns using certain rulesets. Research done on SNORT and Wireshark has shown that the two protocols work hand in hand to provide traffic analysis.

Key benefits of integrating SNORT with Wireshark include:

Real-time Detection: SNORT analyzes packets real time and reports events that look suspicious while wireshark analyzes packet capture data after a certain event. Rule-Based Filtering: SNORT works with different rules define patterns of traffic flow, including signatures of malware or any abnormal traffic on a port. Enhanced Forensics: Due to the opportunity of packet dissection Wireshark can be used to check and explain SNORT alerts in more detail. While this paper only explores Wireshark's packet capture and GeoIP-based geolocation feature of the tool, there is potential for future studies where the tool should be complemented with SNORT for analyzing traffic and correlating geolocation data with intrusion alerts.

2.3 Review Based on the Capability of Wireshark as an Intrusion Detection System

As already mentioned, Wireshark is more of a packet analyzer than an IDS tool Nevertheless, this software has been used in several studies to detect suspicious or even malicious traffic. Wireshark's filtering capabilities and extensive support for network protocols allow it to analyze:

OSC: Other types of encrypted traffic slopes up as SSL/TLS sessions. They include cases of traffic abnormality or packets with wrong header information or packets with wrong payload information. Frequent odor communications with unusual paths, reaching often infrequent or even never before seen places. However, Wireshark's limitations include:

No real-time detection: Unlike SNORT, Wireshark essentially does not offer alerts or program automatic operations regarding inserted patterns. Manual analysis dependency: While analyzing packets with the help of Wireshark, much time and sheer experience could be needed for the accurate results interpretation.

2.4 A Review Based on the Analysis and Design for Network Protocol Analysis System Based on WinPcap

Packet capture is at the core of network traffic analysis, and applications like Wireshark for this use, system libraries like WinPcap (or its updated version Npcap). WinPcap also allows the decoding of traffic on the network interfaces which is otherwise a bit difficult with other usual protocol analyzers, capturing both the inbound and outbound traffic.

Previous studies have emphasized the role of WinPcap in enabling:

Custom packet analyzers: Applications that involve the capturing and analysis of packets, can readily develop custom solutions using the WinPcap APIs. High-speed traffic analysis: WinPcap is designed for high performance that makes it very useful in analysing traffic in high data rate networks. Cross-platform support: Some advantages of the integration of WinPcap are: Iii) WinPcap works with other tools like Wireshark or custom Python scripts.

3 Methods

This section explains the procedures used for monitoring the location of network traffic with Wireshark together with auxiliary tools for data analysis and visualization. In practice, the goal of the method was to specify the origin and the destination of the network packets in a particular environment which might be useful to achieve a range of objectives such as investigating the networks, solving the problems in them or improving the security.

3.1 Identifying Class of IP Address

IP addresses range Table

Class	IP address ranges
A	1.0.0.1 to 126.255.255.254
B	128.1.0.1 to 191.155.255.254
C	192.0.1.1 to 223.255.254.254
D	224.0.0.0 to 239.255.255.255
E	240.0.0.0 to 254.255.255.254

Figure 2: Classes of IP Adress

The first refinement of the extracted IP addresses is the categorization of these IP addresses depending on their functionality and routing characteristics. This step is important to apply the filter on the traffic and to concentrate the geographical localization

on public IPs since they are always globally routable and that provide useful information about the geographic source/destination of the traffic. A packet capture (PCAP) of the Wireshark trace was made and each of the subsequent network packets was divided based on the source and destination IP addresses.

Categories of IP Addresses:

Figure 3: Parts of IP adress

1. Private IPs:

As we know IP addresses designated for private use are intended for usage inside local area networks and cannot be transmitted through the internet. Examples include addresses in the ranges:

192.168.x.x

10.x.x.x

172.16.x.x to 172.31.x.x

These are normally allocated to equipment inside the private network by a router or a DHCP server. Due to the fact that they are used solely for intranet communication within the network they don't go through the geolocation process.

2. Public IPs:

Internet IP addresses are globally reachable and are used as unique network addresses of Internet connected devices. These IPs enable electronic devices to connect to other devices or services offered in the global world. IP addresses offered by Internet Service Providers (ISPs) as limited accessible IP addresses are termed as public IP and are the primary subject of geolocation study. For example, an IP such as 142.250.181.3 which has been classified as a public IP in this work can be mapped geographically to find approximately where that IP is situated physically. They are very useful if one needs insight on the origin or the final point of the incoming/outgoing network traffic.

3. Multicast IPs:

The Multicast IP addresses are used for a group communication model in the system of the IP networks. These addresses enable single sender communicates with multiple receivers at the same time. Examples of multicast IP ranges include:

224.0.0.0 to 239.255.255.255

They are commonly used for specific issues such as video distribution, gaming or Network Discovery Protocols. In the geolocation process, RPKI vali- date only multicast

IPs; therefore, multicast IPs are not assigned with geographical latitudes and longitudes. An address such as 224.0.0.251 will be recognized as multicast and ridiculed from the geo-location scrutiny

3.2 Tracing Geo Location Using Wireshark and GeoIP Database

This paper focuses on using Wireshark to trace the geo-location of a given matrix of IPs with the help of GeoIP database. Track online network flow is one of the significant requirements when determining the source and destination of data packets in cases of outgoing external connections or possible threats. This process was carried out in two stages: monitoring the network traffic using tool Wireshark; identifying visitors' location with the help of the GeoIP database.

Capture of network traffic with a protocol analyzer Wireshark

To capture the network, Wireshark – a professional and popular packet analyzer — was used as a capturing tool. The following steps detail the traffic capture process:

1.Setup and Configuration:

Wireshark tool was deployed and running on the host operating system enabling capture of traffic on a chosen network interface. The type of interface was determined by the activity level or the target analysis of the specific set being processed. An attempt was made to adjust Wireshark filters in case if needed and to exclude unnecessary data capture.

2.Traffic Capture:

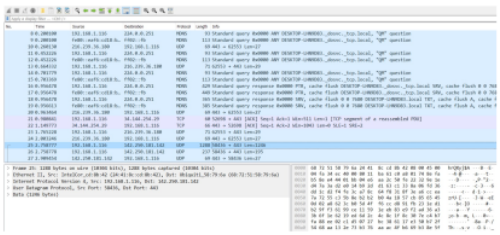

Figure 4: Traffic Capturing

Wireshark captured packets in real-time based on the packets moving in the network segment. The details that each packet included were source and destination IP addresses, protocols, and timestamps. The captured packs included various traffic, including HTTP request, DNS requests, and other types of communication traffic.

3.Data Export:

Having processed the capture, the recorded data was stored in the PCAP (Packet Capture) format under the name wire.pcap. This document is popular and includes all the information needed for further processing.

Implementation in Python

This was done using a python script with dpkt for the PCAP analysis and pygeoip for geolocation. Below is a high-level summary of the process:

Extracting IP Addresses:

The script switched through the packets in wire.pcap file, and 4 and 5 represents the source and destination IP respectively. When carried to the geolocation module, only public IPs were passed.

Querying GeoLiteCity:

For each public IP the GeoLiteCity database was queried and the latitude and longitude was stored in the variable latlon for further processing.

Error Handling:

Where no geolocation data was found possible cases were noted and the script moved on to the next IP address.

Storing Results:

The geolocation results were basically in a format which was organized like a KML file for viewing.

4 Results and Discussion

4.1 Implmentation

Figure 5: Implementation flow diagram

4.2 Results

The analysis results are derived from analyzing the amount of the network traffic recorded in the wire.pcap file. In the analysis, captured packets were subjected to packet sniffing to obtain source and the respective destination IP addresses before conducting geolocation on public IPs. The conclusion of the research is made clear below:

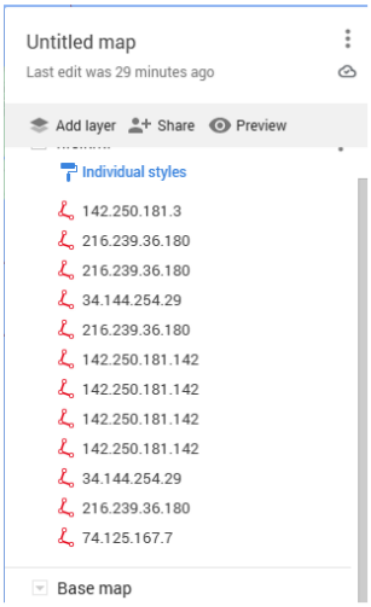

Figure 6: Map parameters displaying IP addresses

IP Extraction Results:

```
Extracted IPs: Source=142.250.181.3, Destination=192.168.1.116
Extracted IPs: Source=142.250.181.3, Destination=192.168.1.116
Extracted IPs: Source=142.250.181.3, Destination=192.168.1.116
Extracted IPs: Source=192.168.1.116, Destination=142.250.181.3
Extracted IPs: Source=192.168.1.116, Destination=216.239.36.180
Extracted IPs: Source=192.168.1.116, Destination=224.0.0.251
Extracted IPs: Source=192.168.1.116, Destination=224.0.0.251
Extracted IPs: Source=216.239.36.180, Destination=192.168.1.116
Extracted IPs: Source=192.168.1.116, Destination=224.0.0.251
Extracted IPs: Source=192.168.1.116, Destination=216.239.36.180
Extracted IPs: Source=192.168.1.116, Destination=224.0.0.251
Extracted IPs: Source=192.168.1.116, Destination=224.0.0.251
Extracted IPs: Source=192.168.1.116, Destination=224.0.0.251
Extracted IPs: Source=216.239.36.180, Destination=192.168.1.116
Extracted IPs: Source=192.168.1.116, Destination=34.144.254.29
Extracted IPs: Source=34.144.254.29, Destination=192.168.1.116
Extracted IPs: Source=192.168.1.116, Destination=216.239.36.180
```

Figure 7: Extracted IP's

These are the IP addresses The extracted **source IPs include:**

1. 142.250.181.3: This IP is classified as the public IP and was later on, traced to its location.

2. 216.239.36.180: This is another INDO and another public IP, which also was geolocated.

3. 34.144.254.29: One belonging to the public IP addresses space connected to a particular place.

Further source IP addresses were obtained but not listed here due to space constrains.

Destination IPs:

These are the IP addresses on which the network packets were arrived. The extracted destination IPs include:

1. 192.168.1.116: A private IP, is an individual IP belonging to a device on a local network.

2. 224.0.0.251: A multicast IP used in group communication.

3 .74.125.167.7: An IP address that could be traced to a particular geographic location.

Public IP Geolocation Results:

The public IPs extracted from the captured data were geolocated using the GeoLiteCity database, yielding the following results:

1. 142.250.181.3:

Geolocation: This IP was found to belong to San Francisco, California. Coordinates: Latitude = 37.4192, Longitude = - 122.0574. Details: This IP is likely to be linked with Google's machines possibly managing search engine, Google applications or similar traffic. It is based in San Francisco which is consistent with Google known infrastructural location in California.

2. 216.239.36.180:

Geolocation: Also originated from San Francisco, California. Coordinates: Latitude: 37.8342, Longitude: -122.2897. Details: Same as the above IP address is associated with Google, probably referring to a different server or node of service. It is one of the same infrastructure or data center near the coordinates 142.250.181.3.

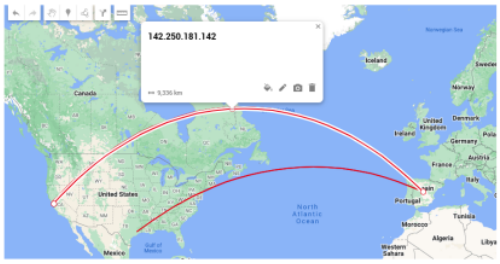

Figure 8: Customization of the map.

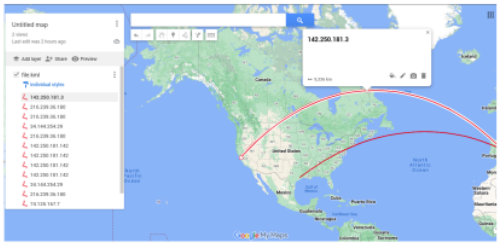

Figure 9: Final output

5 Future Scope

The opportunity to extend the concept of Wireshark traffic virtualization in the future and, in connection with cloud environments, further develop a program of the flagship research in the field of networks. Since the network infrastructures are becoming more and more complex due to multiplicity of nodes and bandwidths and with introduction of cloud services technologies, there will be a higher requirement for improved, effective and elastic tool to detect, analyze and control the network flows. The enhancement of the capabilities offered by Wireshark, as well as its harmonization with other related tools and solutions for network monitoring and security, might greatly expand its uses in intricate and rapidly evolving clouds.

One of the areas that require improvement is the sequence of capture and display filters in Wireshark. These filters are essential when the packets of the network need to be compartmentalized, according to their nature, protocol and use, which are all very important in a network. If better capture filters were to emerge it would be even easier to filter out specific protocols, IPs, or even other attributes of traffic. Furthermore, adding the algorithmic flexibility of using adaptive filtering mechanisms that adapt to the type and load of the offered traffic would increase the stability of the Wireshark toolkit and enable it to address more complex and diverse traffic patterns efficiently.

Furthermore, it can be suggested that, with evolution of the networks' speed, the successive versions of Wireshark can easily be designed to capture the incoming as well as outgoing traffic with high transmission speeds in real time and analyze them. Consequently, enhancements in the packet capture and analysis rate would make explicit sense in today's advanced cloud environments that rely on high-speed connectivity. Wireshark could be extended to operate with 5G networks, IoT traffic and edge computing where the pressures on networks are rising considerably. With such possibilities Wireshark can develop into the most efficient and easy to use tool in the field of the real time traffic analysis and, in particular, diagnostic in the actively developing and, frequently, high-speed networks.

6 conclusion

. Summing up, the findings of this research prove Wireshark as multipurpose packet sniffer for capturing traffic on the network along with utilizing the features of the GeoIP database for IP IP addresses localization. The paper focuses on understanding the extent

to which Wireshark can be used in analyzing the network activity to give a detailed understanding on the network flow of the devices in the network. Since Wireshark captures packets and can filter them the tool successfully identifies the source and destination IP addresses, which are indispensable in analyzing the interactions in the network. These interactions, once passed through the GeoIP interface to identify where external traffic is coming from, provides yet another layer of information about the network's activity.

Wireshark incorporating GeoIP geolocation means that a routine packet capture and analysis can be correlated with geographic origin of the packet magnifying the function of the tool by providing insight into not only what was sent/received form the network layer but also where it came from. The working with the GeoLiteCity database helped in precise linking of given public IP addresses to geographic locations, thus identifying traffic patters that could have otherwise been not observed. These patterns are significant in tracking down valid traffic, mapping suspicious activities and to some extent, we detect the geography of traffic across the internet.

7 Acknowledgement

On earnest, I am grateful to express my heartfelt thanks to Dr. Zain Siddique, Professor, Department of Computer Science for his valuable guidance, encouragement and his valuable time on this aforesaid project. He has offered his best practices, useful tips, and valuable comments that helped to define the further development of this project. It is with a lot of appreciation to her effort as well as time he used in helping us to be what I we are today.

Assignment

ORIGINALITY REPORT

4%	1%	3%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Sudha Arvind, S. Arvind, Vamshi Kumar Silveri, Govardhan Potey, Parameshwar Nunavath, Ranadheep Podishetty. "Network Traffic Virtualization Using Wireshark and Google Maps", 2023 International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE), 2023 Publication	3%
2	forum.mikrotik.com Internet Source	1%
3	article.sciencepublishinggroup.com Internet Source	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		