** **

**PROJECT PROPOSAL**

ANDROID NETWORK SNIFFER

Bachelor of Computer Science (Digital Systems Security)

**Team SSP19/2B**

*Kan Kar Jun Alvin 5709659*

*Liao Weisheng Wilson 6212591*

*Nyein Soe 5710959*

**Supervisor**

*Mr. Premarajan P*

**Project Website**

[*https://android-network-sniffer.webnode.com/*](https://android-network-sniffer.webnode.com/)

**Contents**

1. Android Network Sniffer

1.1. Objective

1.2. Introduction

2. Network Sniffer

3. How does it differ from other existing network sniffing tools

4. Research Conclusions

5. Implementation(proposal)

6. Project Timeline

7. Roles and Responsibilities

8. Team Members Profiles

9. References

1. **Android Network Sniffer**

**1.1. Objective**

The objective of this project is to create a network sniffer in an Android OS, where the program is able to listen to the network communication and break down the packets to readable format. For instance, a WEP program crack can be implemented to provide a more robust implementation.

**1.2. Introduction**

In general, network sniffing involves using sniffer tools that enable real-time monitoring and analysis of data packets flowing over computer networks. It can be a hardware device or a separate software program or a combination of both. It is also called as packet sniffing, snoop, network probe, packet analyzer, network analyzer, or protocol analyzer. Basically, it examines traffic on the network and takes snapshot copies of the packet data.

**2. Network sniffer**

**Different usage of this sniffer tools**

Based on our research, we have found several packet/network sniffing tools which are currently being used and they are as mentioned below.

* SolarWinds Packet Analysis Bundle
* WireShark
* PRTG Network Monitor
* Steel Central Packet Analyzer
* Tcpdump
* Network Miner
* Kismet
* Fiddler
* EtherApe
* Packet Capture

Each type of tools has their own functionalities and to use some of them, the user will have to pay in order to gain access to the full feature of the respective tools.

**Different sniffer tools**

Based on our research, we have found several types of sniffers and they are as mentioned below.

* *ARP Sniffers:* In ARP sniffing, the network traffic is not sent to the hosts, but it is sent to the ARP cache of both network hosts, which is then forwarded to the network administrator. ARP Sniffers sniff the data when it is in the ARP cache. The ARP sniffer is popular among hackers, as the data captured in the cache allows them to create a map of the IP addresses and their associated MAC addresses. This map allows the hackers to conduct packet-spoofing attacks, search for router-based vulnerabilities and ARP poisoning attacks.
* *IP Sniffers:* IP Sniffers sniff all the data that corresponds with a specific IP address filter. This allows capturing of specific data packets for analysis and diagnosis. This method is also popular among hackers, who use it for stealing data and also for stealing the TCP session. They also use this to create fake TCP sessions, act as a man-in-the-middle and unleash MitM attacks.
* *MAC sniffers:* MAC sniffers sniff all the data that corresponds with a specific MAC address filter.
* *LAN sniffer*s: These are deployed on internal LANs and they have the capability to scan the complete IP range.
* *Protocol sniffers:* These are used to sniff data that are related to the network protocols that are used on the network.
* *Web password sniffers:* Hackers steal http sessions parse them to acquire login credentials, user IDs and passwords. While many websites protect their external facing web pages/websites with [SSL](https://securebox.comodo.com/ssl-sniffing/network-sniffing/key5sk1=fbdf6f7639540dba2706b006f150b765c0a209ec), they do not use SSL or use lesser secure encryption for their internal web pages. Hackers can exploit this vulnerability.

**How it can be used to analyze network**

For instance, SolarWinds is used to monitor network performance and have an overall view of what's going on in your network. What this means, more plainly, is it pays mind to more of the pure motility of the network. Transmission speeds and rates, packet transmission reliability, and even comes pre-configured with a wide variety of visual aids and sharp looking charts to make irregularities easier to spot.

Its counterpart, the Network Analyzer, again with a self-explanatory name, is more focused on the traffic itself. While the Performance Monitor is focused more on the overall view of the network's performance, the Network Analyzer is paying a lot more attention to the network on a more granular level.

In particular this part of the program ferrets out the [bandwidth hogs](http://www.pcwdld.com/bandwidth-monitors-best-free-network-tools) and anomalies, sorted by merit of users, protocols, or applications.

**3. How does it differ from other existing network sniffing tools**

**3.1. WireShark**

WireShark is a windows-based network sniffing tool that show all the network information of the network the device is currently connected to. WireShark is able to provide details like the resolved addresses, protocol hierarchy status, connections, endpoints, packet lengths and the IO graphs.

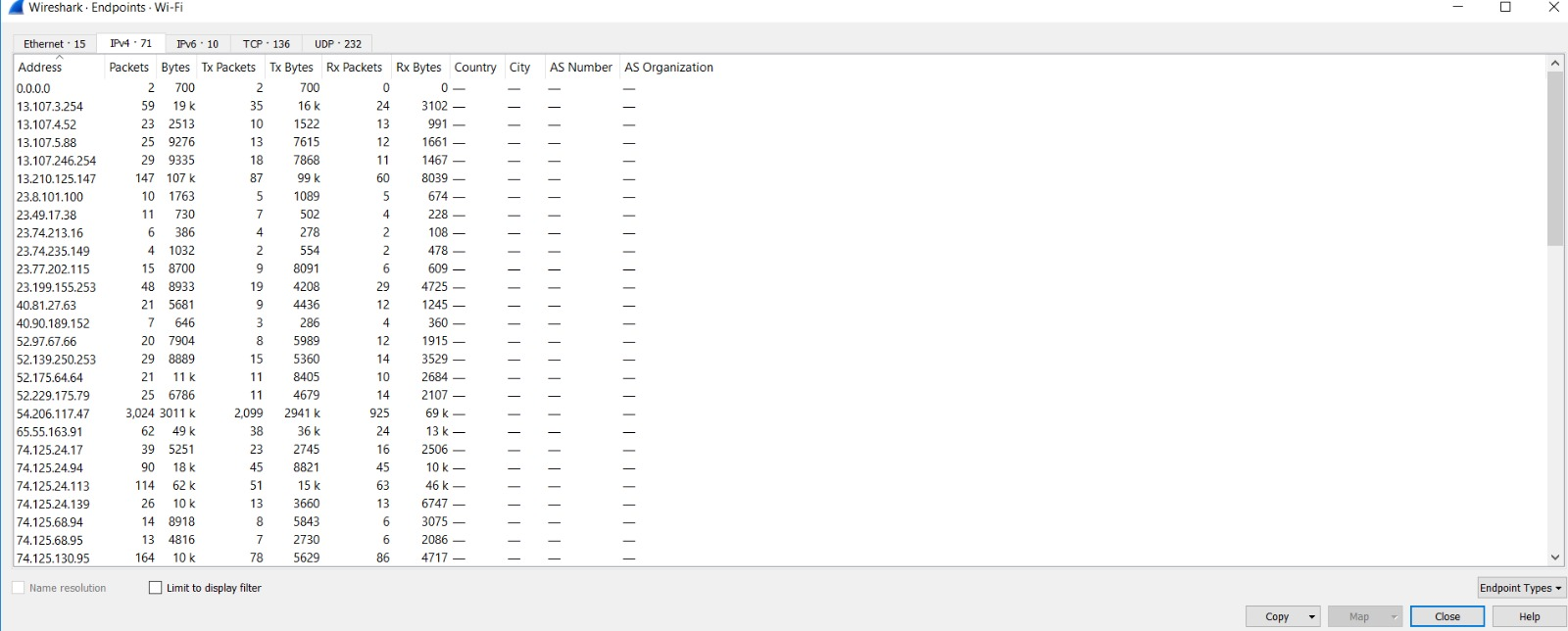


Figure 3.1.1. Features of WireShark

**3.2. PRTG**

PRTG is an android based app that provides information on the devices and services that are connected to the network. These includes information on servers or FTP services that are on the network. Example of these information are the ping, model number, serial number of the devices, and the number of files on the FTP server.

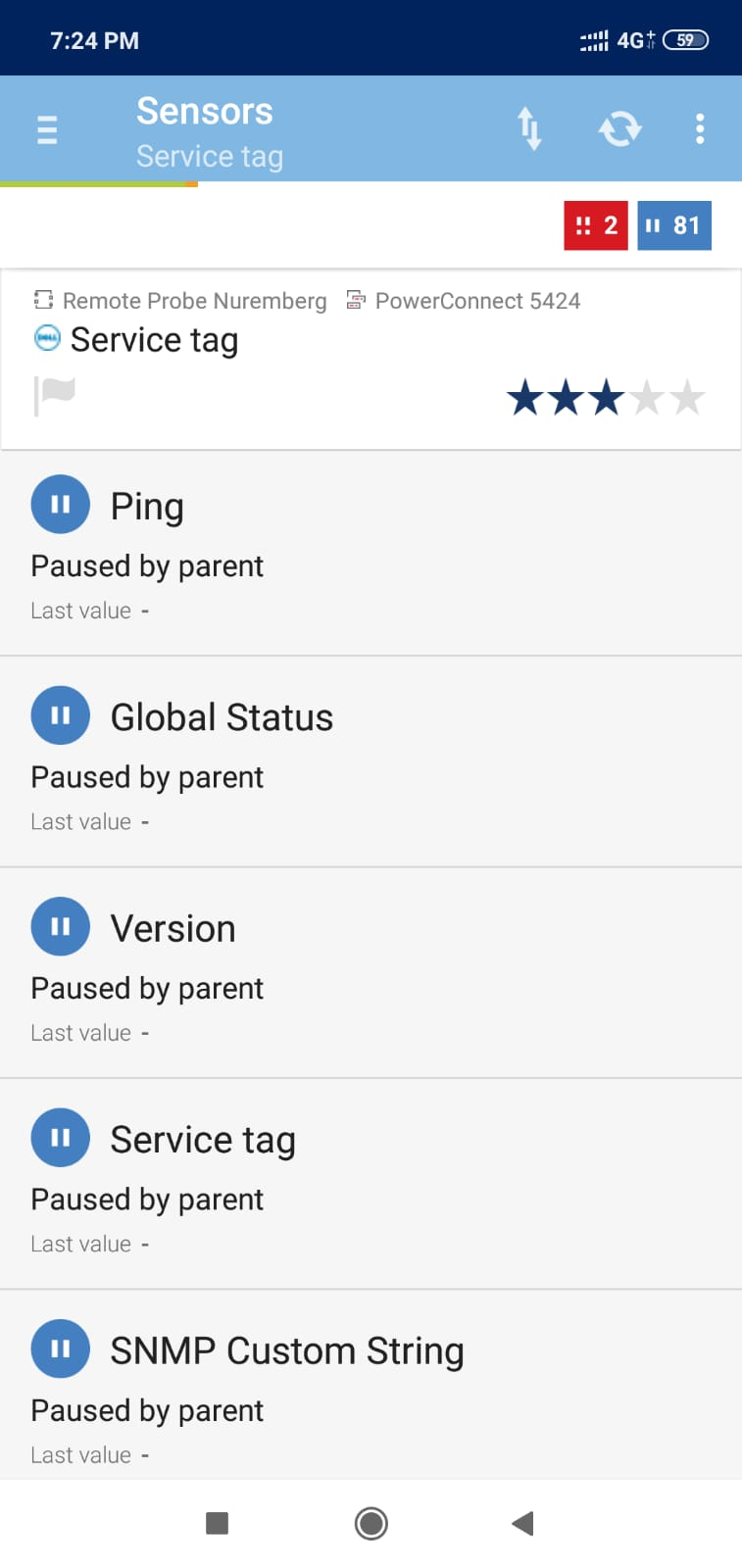


Figure 3.2.1. Features of PRTG

**3.3. Packet Capture**

Packet capture is an android based app that provides information on the usage of the network by the various apps and processes on the device. This information includes the security standards, MAC address, as well as the server details.

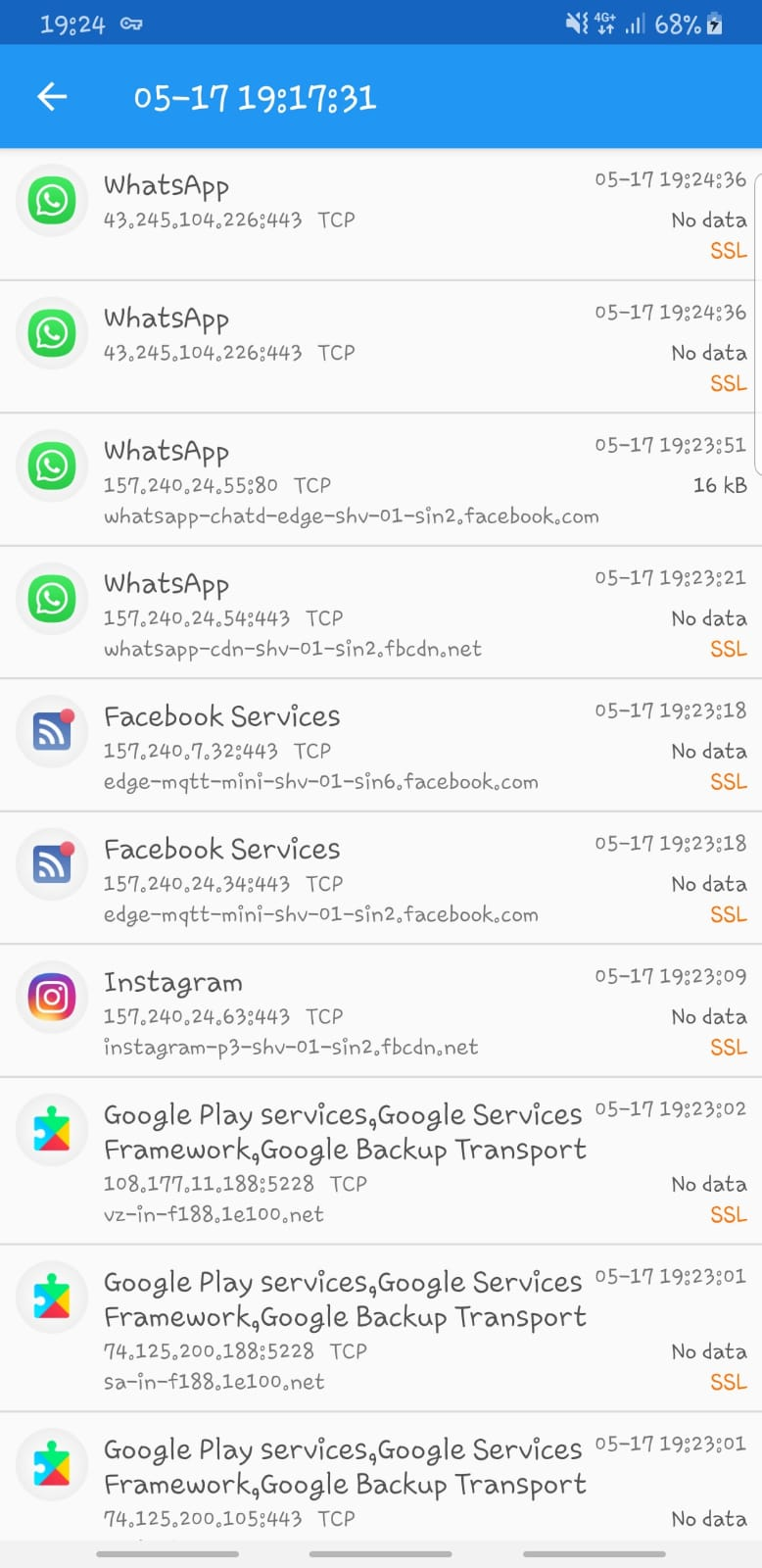


Figure 3.3.1. Features of Packet Capture

**3.4. Network Miner**

Network Miner is also a windows-based network sniffing tool that shows the packet information of the network that the device is connected to. It provides information like the packets sent, these packets show associated packets and if there are ongoing sessions as well as the packet size.

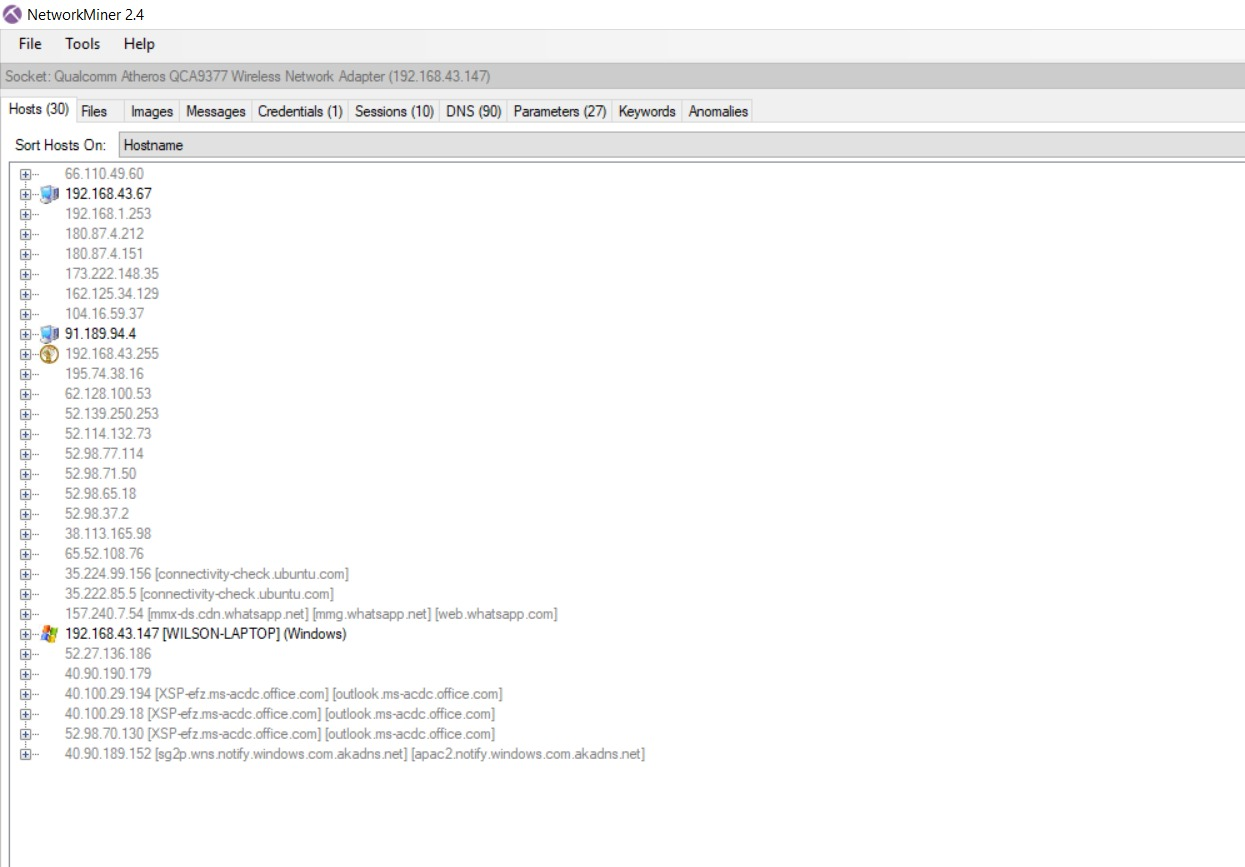


Figure 3.4.1. Features of Network Miner

**4. Research Conclusions**

**5. Implementation(proposal)**

**6. Project Timeline**

**7. Roles and Responsibilities**

|  |  |
| --- | --- |
| **Name** | **Roles/Responsibilities** |
| NYEIN SOE | Team Leader, Researcher, Implementation Designer, Tester |
| LIAO WEISHENG WILSON | Lead Developer, Implementation Designer, Tester |
| KAN KAR JUN ALVIN | Backup Developer, Implementation Designer, Planner, Technical Writer |

**8. Team Members Profiles**

|  |  |  |  |
| --- | --- | --- | --- |
| **Student Number** | **Name** | **Phone Number** | **Email Address** |
| 5710959 | NYEIN SOE | 90721870 | nyeinsoe.wayne@gmail.com |
| 6212591 | LIAO WEISHENG WILSON | 97591961 | z.irashi.z@gmail.com |
| 5709659 | KAN KAR JUN ALVIN | 98346884 | xalvin.k@gmail.com |

**9. References**