SHELL SHARE

A Seminar Report

Submitted By

Lucky Pathan S.

210303105790

Swet Patel R.

210303105787

Vasu Rakholiya D.

210303105794

in Partial Fulfilment For the Award of

the Degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

Under the Guidance of

SHERYA DHOLARIA

Department of Computer Science & Engineering

Parul University

VADODARA

January - 2023

PARUL UNIVERSITY

CERTIFICATE

This is to Certify that PROJECT -2-Subject code 203105400 of 7th Semester entitled “Shell Share” of Group No. PIET_CSE_2 has been successfully completed by

- LUCKY PATHAN S. – 210303105790
- VASU RAKHOLIYA D. – 210303105794
- SWET PATEL R. – 210303105787

under my guidance in partial fulfillment of the Bachelor of Technology (B.TECH) in Computer Science and Engineering of Parul University in Academic Year 2022- 2023.

Date of Submission :-

Project Guide

Prof. Shreya Dholaria,

Head of Department,

Dr. Amit Barve

Project Coordinator:-

Prof. Yatin Shukla

External Examiner

Acknowledgements

"The single greatest cause of happiness is gratitude."

-Auliq-Ice

We would like to express our profound gratitude to **Dr. Amit Barve** (HOD) , of CSE department, and **Dr. Vipul Vekariya** (Dean) of PIET (Parul Institute of Engineering and Technology) for their contributions to the completion of my project title **SHELL SHARE**.

We would like to express our special thanks to our mentor Asst. Prof. **Shreya Dholaria** for her time and efforts she provided throughout the year. Your useful advice and suggestions were really helpful to us during the project's completion. In this aspect, We are eternally grateful to you.

Place: **Vadodara** **Lucky Pathan S. (210303105790)**

Swet Patel R. (210303105787)

Vasu Raknoliya D. (210303105794)

Date:

Abstract

As we all know, current cybersecurity crimes are increasing daily. We need to create a faster way to test the system and secure it. Our tool, SHELL SHARE, will help the company's cybersecurity team coordinate more effectively. This tool allows a team to connect on a single network, enabling them to share their progress, any bugs, files, and a log of every step of each user. If a member successfully breaches the system, they can share the details with other group members. After a successful breach, the log files can be retrieved for better bug documentation. We cannot rely on automation tools like NESSUS, so to facilitate manual testing, SHELL SHARE can be deployed. It's a Python-based tool that can work on Linux systems with certain dependencies. We create the first user as a server host and use this device's IP address and PORT with a random password to allow other members to connect. The user can connect to the server with a command specifying two parameters (IP, PORT, and Password). When a user connects, the host will receive a notification of the user's IP, and they can either allow or deny the connection request.

Keywords: Server Creation,Log File Generation, LAN networks, Manual Testing of Network

Table of Contents

Acknowledgements **iii**

Abstract **iv**

Table of Contents **v**

List of Tables **vii**

List of Figures **viii**

1 Introduction **1**

1.1 **Problem statement** 1

1.2 **Scope** 1

1.3 **Aim and Objectives** 1

2 Literature Survey **3**

2.1 Research Paper Evaluation Table 3

2.2 Critical Evaluation of Journal paper 8

3 Project Flow and Methodology **30**

3.1 Technologies 30

3.1.1 Python 30

3.1.2 Command-Line Interface 31

3.1.3 Custom Server Creation 34
3.2 Project Flow 36
3.3 TimeLine chart 36
4 Future Work **37**
References **38**

List of Tables

2.1 Research Summery Table 8

List of Figures

3.1 Custom Command Parser. 30

3.2 Command-Line Interface. 33

3.3 Create Commond 35

3.4 Connect Commond 35

3.5 Project Flow Diagram. 36

3.6 Time Line Chart. 36

Chapter 1

Introduction

Our project focuses on cybersecurity since manual testing in the company is time-consuming and burdensome for employees. Our tool, SHELL SHARE, aims to streamline this process by allowing users to collaborate within a network, facilitating bug identification. When a user discovers a bug, it will be logged, and other members in the network will be promptly informed of the bug's detection.

1.1 Problem statement

As we know, manual testing of the network can impose a significant workload on the cybersecurity team. Therefore, we need a system that can provide proper bug management and track the employees currently working on a specific bug. When an employee finds a bug, it can be challenging to backtrack the steps and gather the necessary data for the bug report. When a group of team members works on bug finding, it's easy to encounter overlap on the same network segment. This results in a waste of both their time. We need a way to efficiently connect with the team and keep them informed about newly discovered bugs.

1.2 Scope

All companies need a cybersecurity team to secure their digital assets and maintain a strong and secure website. Our tool will assist the team in better communication and automatically gather the necessary information from the log files needed to generate a bug report.

1.3 Aim and Objectives

The aim of our project is to create a Python tool that can assist cybersecurity teams in clearing a network with minimal hassle and providing a super easy way to communicate with other group members. First, we need to create a manual command-line code for the special commands that we will implement in this project to facilitate report generation and creating a list of found bugs.

Chapter 2

Literature Survey

2.1 Research Paper Evaluation Table

Sr. No	Author	Title	Dataset	Method	Result	Future Work
01	Nwauba Nnaemeka Kennedy	Design And Implementation Of Network Security	Bank Network Packet	Basic encryption network with host identification	Ensure bank Network & Security	Add higher encryption
02	Lokesh Rao	Active Network MonitorNet Tracer	Network error rate	Network monitoring tool	Clear bottlenecks in the existing networks	Enhance Network scanning

03	Pietro Grandinetti	Server deployment with Python	-	Operations to get the server a manual config	Create server with host data management	Make a custom server for request and response
04	Tshepang Lekhonkhobe	Argparse Tutorial Python	Python Library	command line options, arguments and sub commands	Understanding the components of the Argparse	-
05	Nathan Jennings	Socket Programming in Python (Guide)	-	Socket Programming	Create server with custom request perimeters.	-
06	Adeyinka, Victor, Adebiyi, Winifred, Olaitan	Packet sniffer for user end network performance monitoring using python	-	Promiscuous mode, TCP/IP header	Dada is being transmitted in form of packets with several no of failure.	Add encryption on files transfer

07	Yaser Mowlaiwzadah	Analyzing and simplifying log files using python	-	FN Log Analysis tool, YM log Analyzer tool ,GUI tool	Program successfully picking log files and these files daily stored in system .	Build database to store the log file
08	Asswini Swain, Het Sheth, Pratik kanani	Remote method invocation using python	PYRO Library	RMI model, RPC tool	Complete end to end RMI system, Successfully implemented RMI in python	Cross network connection
09	V.Neethidevan, G.Chandrasekaran	Web Automation with Python	Selenium Library	Selenium web driver, Web automation	GUI testing for web based application, cross browsers testing in different browsers	Window components handling

10	Aditi Shrivastava, Nitin Shukla	Extracting Knowledge from user access logs	-	Server log files, web usage mining tool	Program successfully tracking logs files of user, 9 of 41 extract information of users with date and time	-
11	Hataw Jalal Mohammed, Kamaran Faraj	A Python wSGI and PHP-Apache Web Server Performance Analysis by Search Page Generator	Search Page Database	Python, Server Analysis	New modifications have occurred in web servers	Improving response time
12	Xia Qing	Network Security Management Platform System Design and Implementation	-	Data encryption, Access control, Data backup, Antivirus	To ensure data and system security in open network environment	Event collection data set improvement

13	Liu Zhijun	Computer network security virtual experiment system design and implementation	SQL database, system design plans	VPN, IDS	Construct platform for network attack and defense	Create more robust system design
14	HUANG Zhikun	The Design and Implementation of Security Network System Based on Web	Database server, Web server	ADO.Net	Cfrom the network between the transport layer and application layer, designed a network security	Create a VLAN for large deployment

15	Dabin Sun, Bowei Wang	Research on the Design of the Implementation Plan of Network Security Level Protection of Information Security	Network Traffic Logs	scheme design, Python,	the correct level security protection level protection design, and improve the security awareness and management level	Coordination and cooperation between departments and relevant personnel at all levels to promote the construction

Table 2.1: Research Summery Table

2.2 Critical Evaluation of Journal paper

Paper 1 : Design And Implementation Of Network Security

Student Name :	Lucky Pathan S.		
Enrollment No :	210303105790	**Branch :**	CSE
Title of Journal Paper:	Design And Implementation Of Network Security		
Authors :	Nwauba Nnaemeka Kennedy		
Journal / Conference :	Department Of Computer Science And Information Technology, Caritas University		
Volume / Issue :	1	**Pages :**	3

Purpose of the research paper is to improve the security that flexible communication infrastructures which provide a diverse set of operations. Understanding the network data flow to manage data packages. Access control is the ability to permit or deny the use of a particular resource by a particular entity. Access control mechanisms can be used in managing physical resources, logical resources, or digital resources. In this document authors mention bank low level network security and plans to improve it by adding encryption and access control for user verification. Bank has no security from attacks like eavesdropping, loops, and 11 of 41 traffic amplification. To overcome from this problem we uses challenge-responses and erasure-coding, for minimum security of the bank network.

In this project , Administration is an aspect of running the organization by devising systems which will run smoothly. The goal of authentication is to first verify that the user, either a person or system, which is attempting to interact with your system is allowed to do so. Gathering basic host information, such as its location and security aspects of its connection, is critical. After all process done, inputs are stripped off with headers and all output is the information which is inside the headers and outputs are in human readable form. Several strategies are implemented like User id and Password , Physical security device, Biometric Identification and more. Based on research we use the network creation and network monitoring data to get multiple hosts with their ID's and User Profiles.

In computer security, access control includes authentication, authorization and audit. It also includes measures such as physical devices, including biometric scans and metal locks, hidden paths, digital signatures, encryption, social barriers, and monitoring by humans and automated systems. Identification and authentication determine who can log on to a system, and the association of users with the software subjects that they are able to control as a result of logging in. Authorization determines what a subject can do. Accountability identifies what a subject (or all subjects associated with a user) did. Based on this research paper , authors described Network Security is essential part of organization.

Critical Analysis:

It is concerned with people trying to access remote services that they are not authorized to use. Most security problems are intentionally caused by malicious people trying to gain some benefit, get attention, or to harm someone. The main purpose of this project is to design a NETWORK SECURITY that will assist bank in the area of ensuring effective security measures. A security context is the collection of roles that a user is associated with. The security context is often defined as part of the authentication process. Depending on the technology used a security context is maintained by the system. Credential is presented to a reader, the reader sends the credential's information, usually a number, to a control panel, a highly reliable processor .Credentials can be passed around, thus subverting the access control list

Paper 2 : Active Network Monitor/Net Tracer

Student Name :	Lucky Pathan S.		
Enrollment No :	210303105790	**Branch :**	CSE
Title of Journal Paper:	Active Network Monitor/Net Tracer		
Authors :	Lokesh Rao		
Journal / Conference :	BSC-IT Project Demonstration		
Volume / Issue :	1	**Pages :**	3

The purpose of the research paper is to monitor the router using network monitoring tools to assist the Network Administrator in identifying bottlenecks in existing networks and measuring error rates in communication, if they exist, and notifying the Administrator. Network monitoring involves gathering essential features and characteristics of the network, such as the number of interfaces connected to the network, incoming and outgoing data bytes, bandwidth, CPU utilization, memory utilization, system uptime, and more. It aids in assessing communication error rates, if any, and informing the Administrator.

In this project, we run a network monitoring tool to monitor the routers. We gather basic network information, such as its users and the number of interfaces. After completing all processes, we can allocate more data bandwidth to the most used interface to accommodate their data requirements. Based on research, we use the network monitoring tool to generate logs for users and document each user's actions along with all possible outcomes they may encounter.

Critical Analysis:

The aim of the project is to assist the Network Administrator in identifying bottlenecks in the existing networks. Network monitoring involves gathering basic features and characteristics of the network, such as the number of interfaces connected to the network, incoming and outgoing data bytes, bandwidth, CPU utilization, memory utilization, system uptime, etc. It helps us measure the error rate in communication, if present, and informs the Administrator. In this research paper, the authors describe the network as an analyzer using a network monitoring tool to make assumptions about network bottlenecks. These tools collect basic features and characteristics of the network, including the number of interfaces, incoming and outgoing data bytes, bandwidth, CPU utilization, memory utilization, system uptime, etc. By using this tool, we can capture user notations, trace user data, and also create log files for users.

Paper 3 : Server deployment with Python

Student Name :	Lucky Pathan S.		
Enrollment No :	210303105790	**Branch :**	CSE
Title of Journal Paper:	Server deployment with Python		
Authors :	Pietro Grandinetti		
Journal / Conference :	Data Science & Optimization		
Volume / Issue :	1	**Pages :**	24

The purpose of this research paper is to demonstrate how to configure a server to run a web application using only Python, without relying on any other tools. We aim to provide insights into the components required for configuring a server for web deployment. We have acquired a reproducible Python template available via GitHub gists. It is essential for a developer to undergo manual deployment on a vanilla server at least once in their lifetime. By spinning up a vanilla machine on any cloud provider and performing a full-fledged production deployment, you will gain a deep understanding of the processes occurring on a remote machine. Knowledge always plays a crucial role.

In this project, the author divides the steps into three parts: Step 1 - Basic machine configuration, Step 2 - Installing the application, and Step 3 - Running the app. Based on research, we utilize the components of server configuration for our server deployment, facilitating host connections. We can create a server with all the essential operations to manually configure the server according to preferred request/response results. Additionally, we can track each user's steps and data on the network to generate logs.

Critical Analysis:

As the author receives the setup explanation, he breaks it down into three steps, each of which is explained briefly. Afterward, we create a detailed view to modify requests and manipulate the host’s throughput while handling all aspects of data flow management. We then create a server with a basic Flask web deployment template, utilizing Python for deployment and managing it solely with Python libraries.

Paper 4 : Argparse Tutorial Python

Student Name :	Lucky Pathan S.		
Enrollment No :	210303105790	**Branch :**	CSE
Title of Journal Paper:	Socket Programming in Python (Guide)		
Authors :	Nathan Jennings		
Journal / Conference :	Python Guide / web-dev		
Volume / Issue :	1	**Pages :**	68

The purpose of this research paper is to demonstrate how to create a simple socket server and client, as well as an improved version that can handle multiple connections simultaneously. We aim to build a server-client application that functions like a full-fledged socket application, complete with its own custom header and content. We will explore the low-level socket API in Python’s socket module and see how it can be used to develop client-server applications. Additionally, we will construct a client and server capable of handling multiple connections using a selectors object. Moreover, we will create our custom class and use it as an application-layer protocol to facilitate the exchange of messages and data between endpoints.

This section covers the primary socket API functions and methods. To create a socket object, you use socket.socket(), specifying the socket type as socket.SOCK_STREAM. The default protocol used is the Transmission Control Protocol (TCP), which is a good default choice due to its reliability and in-order data delivery. When utilizing the loopback interface (IPv4 address 127.0.0.1 or IPv6 address ::1), data never leaves the host or connects to the external network. In the diagram above, the loopback interface is contained within the host, emphasizing its internal nature and indicating that connections and data transmitted through it are local to the host. This is why the loopback interface and IP address 127.0.0.1 or ::1 are often referred to as ’localhost.’”

Critical Analysis:

In this report, the author teaches us that the Python interface is a straightforward transliteration of the Unix system call and library interface for sockets into Python's object-oriented style. The socket() function returns a socket object whose methods implement various socket system calls. Parameter types are somewhat higher-level than in the C interface. Similar to read() and write() operations on Python files, buffer allocation on receive operations is automatic, and buffer length is implicit on send operations.Based on research, we utilize the Multi-Connection Client and Server feature of the socket API, which helps us connect users in a LAN network. We have covered a lot of ground in this tutorial. Networking and sockets are extensive subjects, and if you are new to networking or sockets, there are many pieces to become familiar with in order to understand how everything works together. However, just like Python, it starts to make more sense as you become acquainted with the individual pieces and spend more time with them.

Paper 5 : Socket Programming in Python (Guide)

Student Name :	Lucky Pathan S.		
Enrollment No :	210303105790	**Branch :**	CSE
Title of Journal Paper:	Socket Programming in Python (Guide)		
Authors :	DNathan Jennings		
Journal / Conference :	Python Guide / web-dev		
Volume / Issue :	3.2	**Pages :**	17

The purpose of this research paper is to explore the parser for command-line options, arguments, and subcommands. It aims to provide an understanding of the argparse package, covering topics such as positional arguments, optional arguments, short options, combining positional and optional arguments, and various combinations of these elements. This tutorial serves as a gentle introduction to argparse, the recommended command-line parsing module in the Python standard library. It's worth noting that there are two other modules that serve a similar purpose, namely getopt (equivalent to getopt() in the C language) and the deprecated optparse. Additionally, argparse is based on optparse, making it very similar in terms of usage.

In this report, the author teaches us the basic features of the Argparse package. They demonstrate how powerful this tool is for capturing host command inputs with precise variable matching and using them to execute simple terminal commands with user-defined parameters. Additionally, this package provides a data structure that allows the commands input into the terminal to be used for log file creation. Based on research, we utilize the components of Argparse to create custom command lines for easier host privileges.

Critical Analysis:

Explaining the Python package Argparse, with its super handy features and basic operations, to help programmers smoothly create custom commands and parameters. The argparse module offers more functionality than what is covered in this document. Its documentation is detailed and comprehensive, filled with examples. After completing this tutorial, you should be able to navigate the documentation without feeling overwhelmed. We will learn about the Python package Argparse, covering all its fundamental options, including positional arguments, optional arguments, short options, combining positional and optional arguments, delving into more advanced topics such as using verbosity levels to display additional text, and handling conflicting options.

Paper 6 : : Packet Sniffer for Users End Network Performance Monitoring using Python Programming

Student Name :	Rakholiya Vasu D.		
Enrollment No :	210303105794	**Branch :**	CSE
Title of Journal Paper:	Packet Sniffer for Users End Network Performance Monitoring using Python Programming		
Authors :	Adeyinka A. Adewale, Victor O. Matthews, Adebiyi A. Adelakun, Winifred Amase,Olaitan Alashir		
Journal / Conference :	International Journal of Current Trends in Engineering & Research (IJCTE)		
Volume / Issue :	4	**Pages :**	11

The purpose of this research paper is to explore the parser for command-line options, arguments, and subcommands. It aims to provide an understanding of the argparse package, covering topics such as positional arguments, optional arguments, short options, combining positional and optional arguments, and various combinations of these elements. This tutorial serves as a gentle introduction to argparse, the recommended command-line parsing module in the Python standard library. It's worth noting that there are two other modules that serve a similar purpose, namely getopt (equivalent to getopt() in the C language) and the deprecated optparse. Additionally, argparse is based on optparse, making it very similar in terms of usage.

In this project, we first obtain all raw packets from the network and store them in a buffer. Afterward, the buffer is stripped of various packet headers, including TCP/IP headers. Once the packets arrive, they are initially analyzed, and their output is updated with the latest information obtained in subsequent processes. After completing all processes, the inputs are stripped of headers, and all output consists of the information contained within those headers, presented in a human-readable form.Based on our research, we use promiscuous mode to track packets from the internet using MAC addresses, and we employ a buffer to store raw packets from the network. In the results, each packet's sender and receiver addresses, port numbers, and headers are extracted and converted into a human-readable format. During data transmission, one or more packets may fail to reach their destination.

Critical Analysis:

Based on the author's conclusion, a packet sniffer software has been developed using Python and implemented on the Windows operating system, as previous packet software was developed using Python and implemented on the Linux OS. In Windows, the packet sniffer accurately captures data, converts it, and displays it in a proper format, making it easier to read traffic and minimizing navigation difficulties compared to the previous software. The performance is evaluated based on the packet loss ratio and TCP packet loss ratio.According to this research paper, the authors describe how due to the rapid growth in networks and increasing users, networks have become vast and complex. This complexity increases the chances of malicious attacks, making network traffic monitoring crucial to prevent potential disasters. The authors' main aim is to continuously monitor packets transmitted across the entire network. To achieve this, they use promiscuous mode, where only packets with their own MAC addresses are transmitted on the NIC, while all others are discarded. They begin by collecting all raw packets from the network and storing them in a buffer. Afterward, they strip off various types of headers to extract TCP/IP headers, perform analysis, update the output file with the latest information, and convert it into a human-readable format. Based on the implementation, it can be observed that as the total number of packets increases, the chances of losing packets decrease.

Paper 7 : : Analyzing and Simplifying Log Files using Python

Student Name :	Rakholiya Vasu D.		
Enrollment No :	210303105794	**Branch :**	CSE
Title of Journal Paper:	Analyzing and Simplifying Log Files using Python		
Authors :	Yaser Mowlaiwzadah, Master Degree Student ,ECE Department, REVA university, Bangalore , India		
Journal / Conference :	nternational Journal of Engineering Research & Technology (IJERT)		
Volume / Issue :	Vol.9/Issue 05	**Pages :**	4

The objective of this research paper is to simplify and analyze log files using YMlog and an analyzer tool to prevent unauthorized access by third-party users. This research focuses primarily on server-based logs such as DNS (Domain Name Systems), FTP (File Transfer Protocol), DHCP (Dynamic Host Control Protocol), Authentication, and more.In the results, this program has two versions: the Script version, which has no GUI and is used on servers, and the Graphic version, which is designed for desktop users. Using this tool, administrators can easily discern what is happening within the system and recognize the importance of log files in system security.In this document, authors emphasize the significance of log files for users, as they carry sensitive and important data over the internet. These log files play a crucial role in identifying information about attackers and unauthorized access.

This program was developed for the Linux operating system and cannot be used on other operating systems. The method involves two separate files: the GUI file, which serves as the main file for local systems with graphical user interface access, and the server-based file, which contains code for non-GUI environments and server environments where there is no access to GUI interfaces. The latter file functions similarly to the former but lacks a GUI interface.Based on this document, we use two separate files: one for host users, which is the GUI file, allowing network hosts to view and observe logs, and another file, the non-GUI file, used for server-side log tracking.

This program was developed for the Linux operating system and cannot be used on other operating systems. The method involves two separate files: the GUI file, which serves as the main file for local systems with graphical user interface access, and the server-based file, which contains code for non-GUI environments and server environments where there is no access to GUI interfaces. The latter file functions similarly to the former but lacks a GUI interface.Based on this document, we

use two separate files: one for host users, which is the GUI file, allowing network hosts to view and observe logs, and another file, the non-GUI file, used for server-side log tracking.

Critical Analysis:

In this documentation, the authors mention why log files are important for any user, as log files carry sensitive data that travels on the internet. Using log files, administrators can identify any attacks or unauthorized access occurring on the network. The authors also discuss various types of log tools they have created, explaining how these tools work and their drawbacks. It's important to note that these tools, developed by the authors, can only function on Linux operating systems. The tool is divided into two parts: the local files (GUI files) are used on the client-side by administrators to monitor and analyze log files on the network through different ports. The server-side is employed to identify other users' logs within the host network, using user IP, username, and password. There are pre-designed pages intended for future work.

Paper 8 : : Remote Method Invocation Using Python

Student Name :	Rakholiya Vasu D.		
Enrollment No :	210303105794	**Branch :**	CSE
Title of Journal Paper:	Remote Method Invocation Using Python		
Authors :	Ashwini Swain , Het Sheth , Pratik Kanani		
Journal / Conference :	NTERNATIONAL JOURNAL OF ADVANCED STUDIES IN COMPUTER SCIENCE AND ENGINEERING,IJASCSE (IJERT)		
Volume / Issue :	Vol.6/issue 11	**Pages :**	4

The objective of this document is distributed computing, which is widely used to facilitate efficient storage and transmission of data over the network. With distributed computing, we can transmit data over high-speed networks in a shorter amount of time. In this system, a single system is connected to multiple computers through a high-speed network. The main idea is to send and receive data quickly, achieving better throughput compared to a single-processor system.In this research, we explore RPC (Remote Procedure Call). The RPC model is similar to the LPC (Local Procedure Call) model, as both are used to transfer and control data within a program. In this document, we focus on RPC, although it uses a structured programming approach that can be improved by using object-oriented models. With this approach, objects in different procedures can communicate with each other through RMI (Remote Method Invocation). We implement RMI in Python, leveraging the advantages of Python, such as readability and the ability to build it with a limited number of lines of code.

In this research, we implemented RMI using socket programming. Initially, user requests are converted into objects, and these objects invoke methods across the network. However, this approach typically requires high-level programming. To streamline this effort, we utilized the PYRO library. PYRO automatically creates objects, locates ports, and enables the creation of an efficient distributed system, allowing many people to connect.Through this research, we employed RMI to establish connections between objects and utilized RPC to transfer and control data over the network. Additionally, we leveraged the PYRO library to simplify implementation, reduce code complexity, and facilitate distributed programming.

In this document, the authors utilize RPC to transfer and control data over the network, while RMI is used to invoke objects across the network, allowing these objects to communicate with each other. The PYRO library is employed to create objects, locate ports, and facilitate distributed programming. Since it's written in Python, it is platform-independent and can function on different operating systems simultaneously. It can also accept any return type of value and method.Using PYRO, users can easily write code and automatically create objects while assigning ports. Objects enable distributed programming, and in this documentation, the authors complete an end-to-end RMI system implementation. Although we cannot directly compare the implementation of RMI in Java with that in Python, given the ever-increasing Python community, Python may eventually supplant Java in the field of distributed programming.

Critical Analysis:

Based on the documentation, the authors clearly explain how various methods and tools can be used to perform distributed programming using Python. In this research, the authors employ different tools such as RPC, RMI, and the PYRO library to create objects and assign them different port numbers. These objects communicate with each other and are responsible for transferring and controlling data across the internet.By using this technique, users can easily transmit data at high speeds within a short amount of time. Additionally, the PYRO library simplifies the process of creating applications that work with RMI. Developers no longer need to engage in high-level coding to create objects and allocate ports, as this is automatically handled by RMI.

Paper 9 : : Web Automation using Selenium Web driver Python

Student Name :	Rakholiya Vasu D.		
Enrollment No :	210303105794	**Branch :**	CSE
Title of Journal Paper:	Web Automation using Selenium Web driver Python		
Authors :	V.Neethidevan, G.Chandrasekaran		
Journal / Conference :	International Journal of Recent Technology and Engineering (IJRTE)		
Volume / Issue :	Vol.7/issue 6s	**Pages :**	3

In this research, the author conducted web automation testing using Selenium WebDriver in Python. Cross-browser testing was performed using various browsers to assess the application's performance as expected. The web-based application underwent testing using Selenium WebDriver with Python code.In this document, the authors build a testing system using Selenium, a tool employed for automated software testing. Selenium drivers allow testers to conduct testing more effectively. In today's context, user interface testing is time-consuming due to the increased number of elements, requiring more time dedication from the development team. To understand user interface testing, testing team members must grasp the various specifications of the system. Testing should commence from day one itself, similar to the involvement of the customer in all phases of the software system. This team should also be engaged from day one.

In this research, the authors conducted cross-browser testing using the Selenium driver. The primary objective of cross-browser testing is to ensure that an application performs its intended operations consistently across different browsers. Every software application needs to undergo multi-browser testing to verify that it functions correctly on various browsers. Testers ensure that components such as AJAX requests, Applets, Flash, etc., work seamlessly on different browsers. From this document, we learned how to use web automation so that our application can operate on any web browser without errors. We employed the Selenium WebDriver to perform automatic testing of our software. In this project, Selenium, an open-source testing tool with a web automation framework, was utilized to automate website testing. It possesses the capability to streamline the testing process, making it more efficient for test engineers.

Critical Analysis:

This paper provides an explanation of web automation and the Selenium software, along with its features and basic operations. Selenium can perform testing on various browsers automatically, eliminating the need for manual testing. It consistently produces effective output across different browsers. The paper describes how to execute GUI testing techniques for web-based applications, highlighting the significance of Selenium WebDriver in Python for web automation. It also emphasizes the importance of cross-browser testing, which was implemented in Chrome and Firefox.Based on the paper, the author discusses the challenges associated with manual testing of software, which often consumes a significant amount of time. To overcome these difficulties, testers utilize the Selenium tool to conduct automated testing on web-based software. Web automation is employed to perform testing in various browsers, ensuring that the application functions effectively and delivers the expected results.

Paper 10 : Python-WSGI and PHP-Apache Web Server Performance Analysis by Search Page Generator

Student Name :	Swet Patel R.		
Enrollment No :	210303105787	**Branch :**	CSE
Title of Journal Paper:	Python-WSGI and PHP-Apache Web Server Performance Analysis by Search Page Generator (SPG)		
Authors :	Hataw Jalal Mohammed , Kamaran Hama Ali Faraj		
Journal / Conference :	University of KURDISTAN Hewler		
Volume / Issue :	Vol.5/Issue 1	**Pages :**	7

The web servers (WSGI-Python) and (PHP-Apache) are in middleware tier architecture. Middleware architecture is between frontend tier and backend tier, otherwise it's a 23 of 41 connection between frontend tier and backend tier for three tier architecture. The eLearning systems are designed by two different dynamic web technologies. First is by PythonWSGI and the second is by Personal Home Page (PHP-Apache). The two websites were designed with different open source and cross platform web technologies programming language namely; Python and PHP in the same structure and weight will evaluate perform over two different operating systems (OSs):

1) Windows-16 and 2) LinuxUbuntu 20.4. Both systems run over the same computer architecture (64bit) as a server side with a common backend MySQL web database for both of them.

The software (SW) and hardware (HW) for web servers and E-management are an essential factor, but this paper concentrates on software and application. Since web application is a crucial part of every functioning web application system designed in different web technologies, Python and PHP with different web applications are WSGI-Python and Apache- PHP and benchmarked. The benchmark has been developed to assist society in its practice of delivering quality improvement (QI). QI for the proposed system is two different web servers (i.e., WSGI and Apache). Other sections in the paper investigate which web server is enhanced and is calculated by handwriting codes for SPG. The enhanced QI illustrates the results in this paper for the mentioned technologies of a search page generator in milliseconds.

Critical Analysis:

WBRP stands for “Web Base Response Performance” which is an aspect of measuring the speed of web technology XAMPP over the different OS. This, in turn, is related to the design and developments, as the faster website is shown to enhance visitor attention, loyalty, and satisfaction for XAMPP, which is currently the widespread web technology application software and support web servers of Apache (Othman et al., 2020). The Web Server Gateway Interface (WSGI) is a standard interface between web servers and Python web application frameworks. By standardizing behavior and communication between web servers and Python web frameworks, WSGI makes it possible to write portable Python web code that can be deployed in any WSGI-compliant web server. WSGI is documented in PEP 3333.

Paper 11 : Extracting Knowledge From User Access Logs

Student Name :	Rakholiya vasu D.		
Enrollment No :	210303105794	**Branch :**	CSE
Title of Journal Paper:	Extracting Knowledge From User Access Logs		
Authors :	Aditi Shrivastava, Nitin Shukla		
Journal / Conference :	International Journal of Scientific and Research Publications		
Volume / Issue :	Vol.7/Issue 4	**Pages :**	4

In today's world, the number of users on the World Wide Web is constantly increasing. It is crucial for website owners to have a deep understanding of their customers to provide enhanced services and improve the overall quality of their websites. To achieve this, they heavily rely on log files. These log files, maintained by web servers, contain valuable information each time a user requests resources from the site. In this study, we delve into web access log files and the information that can be extracted from them, aiding in understanding user behavior. This information is instrumental in the restructuring and redesigning of websites. Accessing information in today's world is one of the most common tasks. Each day, we encounter a plethora of information, influencing both website owners and visitors. Website owners reach a broad audience, nationally and internationally, and provide 24x7 customer service. Concurrently, the number of visitors to websites is increasing rapidly. Data mining techniques are applied to mine information. However, websites are unstructured, making it difficult to directly apply mining techniques. Hence, we employ web mining, which is tailored to work with web data.

”Web usage mining is a research field that focuses on the development of tools used to study users' web navigation behavior. A server log is a logging file that is automatically created and manipulated by a server in real-time. It's a file where the web server writes information each time a user requests resources from a particular site. It represents the activity for every user over a period of time. In this study, we utilize server log files, which are automatically created and manipulated by the server in real-time. We also employ web usage mining to collect log files and gather information about user actions. These files are typically only a few megabytes in size.

The data management subsystem operates on a B/S structure and includes various functions such as integrated display, alarm management, device management, event management, emergency management, report management, system management, and expansion management. These functions interact with the data in the security information database through the security event analysis layer. The data is then presented through page displays and statistical reports.

Critical Analysis:

This document contains server log files, which are used for mining and collecting important user information for each request. Website owners use this data to better understand their customers. They also utilize log analysis to collect data from the server and obtain relevant information. Log files are generated with the date and time of each transaction. They contain the particular client IP addresses that requested the site. When a user logs in, their username is included in the log files. The log files show the different status of particular requests, as well as how many bytes were sent and received from the server, along with the time duration. The web is a critical medium for conducting business and commerce. Therefore, designing web pages is crucial for system administrators and web designers. This design aspect can greatly impact the number of visitors. Hence, web analyzers need to analyze data with server log files to detect patterns. In this study, we aim to understand web server logs and discover information that can be used to improve business performance. The authors of this documentation describe log files, detailing how they are created on websites and their purpose. They also explain server logs and log analysis. Furthermore, they elaborate on why logs are important for website owners and how they use this data to understand their audience and make improvements to their websites, ultimately attracting more customers to join and interact with them. The authors also provide insight into the different types of data a log file can contain, such as user login details, timestamps, server IP, and port. They go on to describe various types of logs in detail.

Paper 12 : Network Security Management Platform System Design and Implementation

Student Name :	Swet Patel R.		
Enrollment No :	210303105787	**Branch :**	CSE
Title of Journal Paper:	Network Security Management Platform System Design and Implementation		
Authors :	Xia Qing, Library, Huaihai Institute of Technology ,Lianyungang, China		
Journal / Conference :	International Conference on Computer Engineering and Technology		
Volume / Issue :	Vol.7/Issue 2	**Pages :**	4

With the rapid development of the Internet age and the increasing popularity of the Internet, network security has become increasingly prominent. Therefore, ensuring data and system security in an open network environment has become a concern for many in the industry. Firewall technology refers to specialized networking equipment used to enhance access control between networks. Its purpose is to prevent external network users from gaining entry into the internal network through illegal means and accessing internal network resources. This protection helps safeguard the internal network operating environment.

The security event collection subsystem utilizes a C/S structure, consisting of two main parts: security event collection and formatting. The first part employs SYSLOG and SNMP protocols to gather security incidents, while the second part involves storing the collected security incidents in the security information database after proper formatting. This serves as the foundational data for the entire network security management platform system.

The data management subsystem operates on a B/S structure and includes various functions such as integrated display, alarm management, device management, event management, emergency management, report management, system management, and expansion management. These functions interact with the data in the security information database through the security event analysis layer. The data is then presented through page displays and statistical reports.

Critical Analysis:

The data management subsystem primarily utilizes the security event information collected by the security event acquisition subsystem. It comprehensively displays information useful to users and carries out configuration management for security devices. It also generates security event reports and performs other functions. The data management subsystem extracts data (security events, among other data) from the security information database to display or stores it in the database after the corresponding operation. The security information database is used to store most of the system data (security events and other data), while a small portion of the system data is stored in XML files.

Paper 13 : Computer network security virtual experiment system design and Implementation

Student Name :	Swet Patel R.		
Enrollment No :	210303105787	**Branch :**	CSE
Title of Journal Paper:	Computer network security virtual experiment system design and implementation		
Authors :	Liu Zhijun, Zhejiang Textile & Fashion Vocational College		
Journal / Conference :	International Conference on Intelligent Computation Technology and Automation		
Volume / Issue :	-	**Pages :**	5

Network security is a critical factor influencing the development of computer networks. In this paper, we focus on studying the virtual experiment system under this backdrop. We have designed and implemented a ’network security’ virtual experiment system on the Visual C++ 6.0 and Visual Studio 2005 platforms. The paper elaborates on the design ideas and implementation methods for each function module of the system, and also outlines a database design methodology. The continuous growth and development of the Internet have significantly enhanced the convenience and benefits for users. However, it has also brought about concerns regarding network security. Ongoing network security incidents pose a significant threat to the rapid progress of the Internet, causing distress among people. Through the computer network security virtual experiment system, users can experience the actual experimental process, understand the perils associated with network security issues, and acquire knowledge about defending against network attacks.

The experiment mainly utilizes a Trojan scanner on this virtual host for Trojan scanning. Subsequently, it individually removes the detected Trojans. During Trojan scanning, not all hosts can yield scan results. These Trojans infiltrate the host through injection attacks, and their removal is only possible after a successful operation. In the database query, only Trojan events consistent with the machine's virtual IP, Trojan ID, Trojan name, and Trojan details are retrieved from the table. The retrieved information includes Trojan ID numbers, names, and the associated harm to the host. When users choose to remove a Trojan, it corresponds to deleting the selected Trojan information from the database. For defending against SYN FLOOD attacks on the attack end, three parameters are available for users to configure: TcpMaxPortsExhausted (TCP connection requests).

Critical Analysis:

The valid values for TcpMaxHalfOpen and TcpMaxHalfOpenRetried fall within the range of 0-65535. A suggested value for TcpMaxPortsExhausted is 5. For TcpMaxHalfOpen, recommended values range from 100 to 65535, with a suggested value of 500. TcpMaxHalfOpenRetried should have values between 80 and 65535, with a recommended value of 400. When any one of these parameters exceeds the specified values, the SYN protection mechanism is initiated. In TCP concurrent connection attack defense, there are two parameters for the experimenter to configure: the threshold and the maximum number of simultaneous connections. The threshold determines the number of TCP connections on the machine; once it reaches this threshold, TCP concurrent connection protection is activated. The system monitors all connections for source IP address, destination address, packet size, and other relevant information. The maximum number of simultaneous connections defines the limit at which the host will no longer accept additional TCP connections from other hosts on the machine. This helps protect the host. The DMZ domain (isolation) encompasses one or more networks. The area where the host or server is located is referred to as a bastion host. Commonly, Web servers and E-mail servers are placed in the isolation zone. This enables network users to access enterprise information publicly, but restricts their access to the protected internal network.

Paper 14 : The Design and Implementation of Security Network System Based on Web

Student Name :	Swet Patel R.		
Enrollment No :	210303105787	**Branch :**	CSE
Title of Journal Paper:	The Design and Implementation of Security Network System Based on Web		
Authors :	HUANG Zhikun1, Wuhan Polytechnic, Hubei Wuhan 430074, China		
Journal / Conference :	IEEE Workshop on Advanced Research and Technology in Industry Applications (WARTIA)		
Volume / Issue :	-	**Pages :**	3

Computer network technology is advancing rapidly, with Internet technology evolving even faster. In this scenario, people are increasingly aware of the importance of network security. Due to the need for security protection in the application of information systems, the study of computer network security consistently revolves around information systems. Presently, global computer network security companies and scientific research departments are actively researching and addressing network security concerns. They have not only developed a variety of hardware and software products for maintaining network security but have also introduced a range of network communication standards and specifications. This article focuses on designing a network security system based on the Web, addressing the layers between transport and application, aiming to create a truly secure Internet network.

With the rapid development of science and technology and the enormous increase in network information, there is a growing demand to increase user access speed to websites. This has led to a heightened focus on database access technology by a larger customer base, seeking to work with it more effectively. In the process of database linking and access technology, ADO.Net has gained increasing trust and praise from enterprises and customers. The security network system structure is a major consideration, encompassing security mechanisms and security objects. The security objects mainly include network security, information security, equipment security, system security, database security, information security, and computer virus prevention and treatment, among others.

Critical Analysis:

To ensure a well-rounded approach to information security and maintain the integrity of network information, the barrel principle of network information security is implemented. The barrel principle addresses security vulnerabilities and threats comprehensively within the system. In terms of prevention, it ensures that each potential weak point in the system can be identified to mitigate risks. This encompasses the design, evaluation, and testing of every part of the information security system. The objective is to thwart common attack methods and fortify the system's security through well-designed security mechanisms and services. Moreover, from a broader perspective, this approach is fundamentally aimed at enhancing the overall system performance by minimizing the safety coefficient. A sound safety plan must be intricately woven into the architecture of a secure network system, as the security architecture forms the bedrock for security scheme design and analysis.

Paper 15 : Research on the Design of the Implementation Plan of Network Security Level Protection of Information Security

Student Name :	Swet Patel R.		
Enrollment No :	210303105787	**Branch :**	CSE
Title of Journal Paper:	Research on the Design of the Implementation Plan of Network Security Level Protection of Information Security		
Authors :	Dabin Sun, Bowei Wang		
Journal / Conference :	International Symposium on Mechatronics and Industrial Informatics (ISMII)		
Volume / Issue :	-	**Pages :**	5

Information security is a widely discussed topic in the era of big data. With the rapid development of computer networks, various information security issues frequently occur. Improving the level of computer network information security protection through the implementation of network security measures is the primary focus of network security. Based on existing security technology and products, combined with the information security protection system, this paper analyzes the relevant technical and management requirements of the scheme design, studying the design methods and processes

We divide the information system into five levels according to the severity of the damage to the social order, public interest, and national security after the information system is damaged. The corresponding information system security level protection is also divided into five levels. The basic requirements for information system security level protection include technical requirements and management requirements. Among them, the technical requirements are divided into five aspects: physical security, network security, host security, application security and information data security. The management requirements include safety management institutions, safety management systems, personnel safety management, system construction and system operation and maintenance management.

Critical Analysis:

The principle of autonomous protection. That is, the user unit of the information system should operate the information system strictly in accordance with national laws and regulations, independently determine the information system security protection level, and organize and carry out the information system security protection work by itself. Secondly, the principle of key protection. That is to say, the information system is classified according to the importance of the information system and the characteristics of the business. Then, different security protection plans are implemented after the implementation of different security protection schemes. This will give priority to protecting the information system of the core business and important key information resources. Thirdly, the principle of simultaneous construction.

Chapter 3

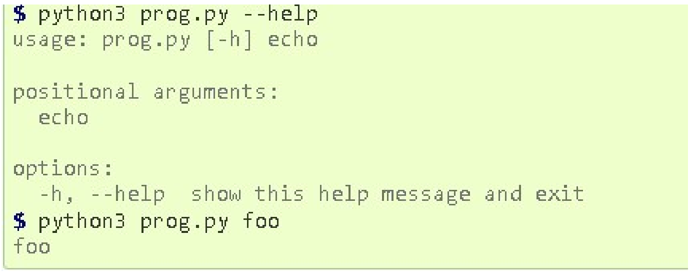
Project Flow and Methodology

3.1 Technologies

3.1.1 Python

Python is an extremely useful programming language for cybersecurity professionals because it can perform a multitude of cybersecurity functions, including malware analysis, scanning, and penetration testing tasks. Python is often recommended as the first language for people new to cybersecurity due to its wide usage and minimal learning curve. We use Python as the core base of the project.

We will create custom user commands to facilitate the understanding of the commands. For the creation of these custom commands, we will use a Python library called Argparse. It’s important to note that there are two other modules that serve a similar purpose: getopt (an equivalent of getopt() from the C language) and the deprecated optparse. Additionally, argparse is based on optparse and is very similar in terms of usage. When we use argparse.ArgumentParser() and parser.parse_args(), running the script without any options will result in nothing being displayed to stdout. However, when we add a parser.add_argument(”echo”) in the command, it will output accordingly.



```
$ python3 prog.py --help
usage: prog.py [-h] echo

positional arguments:
  echo

options:
  -h, --help  show this help message and exit
$ python3 prog.py foo
foo
```

Figure 3.1: Custom Command Parser.

As there will be a command for a new user to create a new LAN server to have the team connected to it. For server creation, we can use the Python library SocketServer. A server is a software that waits for client requests and serves or processes them accordingly. On the other hand, a client is the requester of this service. A client program requests some resources from the server, and the server responds to that request. A socket is the endpoint of a bidirectional communication channel between the server and the client. Sockets may communicate within a process, between processes on the same machine, or between processes on different machines. For any communication with a remote program, we have to connect through a socket port. The main objective of this socket programming is to understand how socket server and client communicate with each other. This program is similar to the server program, except for binding. The main difference between the server and client programs is that in the server program, it needs to bind the host address and port address together. So, we create an INET, streaming socket, socket.socket(socket.AF_INET, socket.SOCK_STREAM). Now to connect to it, we use s.connect((’IP’, PORT)). When the connection completes, the socket s can be used to send a request for the text of the page. The same socket will read the reply and then be destroyed. We used socket.gethostname() so that the socket would be visible to the outside world. If we had used s.bind((’localhost’, 80)) or s.bind((’127.0.0.1’, 80)), we would still have a ’server’ socket, but one that was only visible within the same machine. s.bind((”, 80)) specifies that the socket is reachable by any address the machine happens to have.

3.1.2 Command-Line Interface

The Shell Share CLI is a tool that allows users to connect to a server, create a user, and perform various actions related to server interaction. Users can log actions, view connected users, and exit the program using this CLI.

To use the Shell Share CLI, run the Python script from the command line. Follow the prompts and input the appropriate commands to perform actions such as connecting to a server, creating a user, logging actions, and viewing connected users.

The following commands are available in the Shell Share CLI:

1. **connect [IP] [PORT]** : Connect to a server with the specified IP address and port.

2. **create [port] [password]** : Create a user with the specified port and password.

3. **log [filename]** : Log actions to the specified file.

4. **shell** : View the list of users currently connected to the local server.

5. **help** : Display available commands and their usage.

6. **exit** : Exit the program.

- **Functions and Features of CLI**

1. **get_wifi_ip()**

   - Description: Retrieves the local IP address of the system for a specific range.
   - Returns: The local IP address or None if not found.

2. **connect(args)**

   - Description: Initiates a connection to a server using the provided IP address and port.
   - Parameters:
     - args: A list containing the IP address and port to connect to.
   - Actions:
     - Prompts the user for a password.
     - Starts a connection process in the background using multiprocessing.
     - Checks the status of the connection process and prints the result.

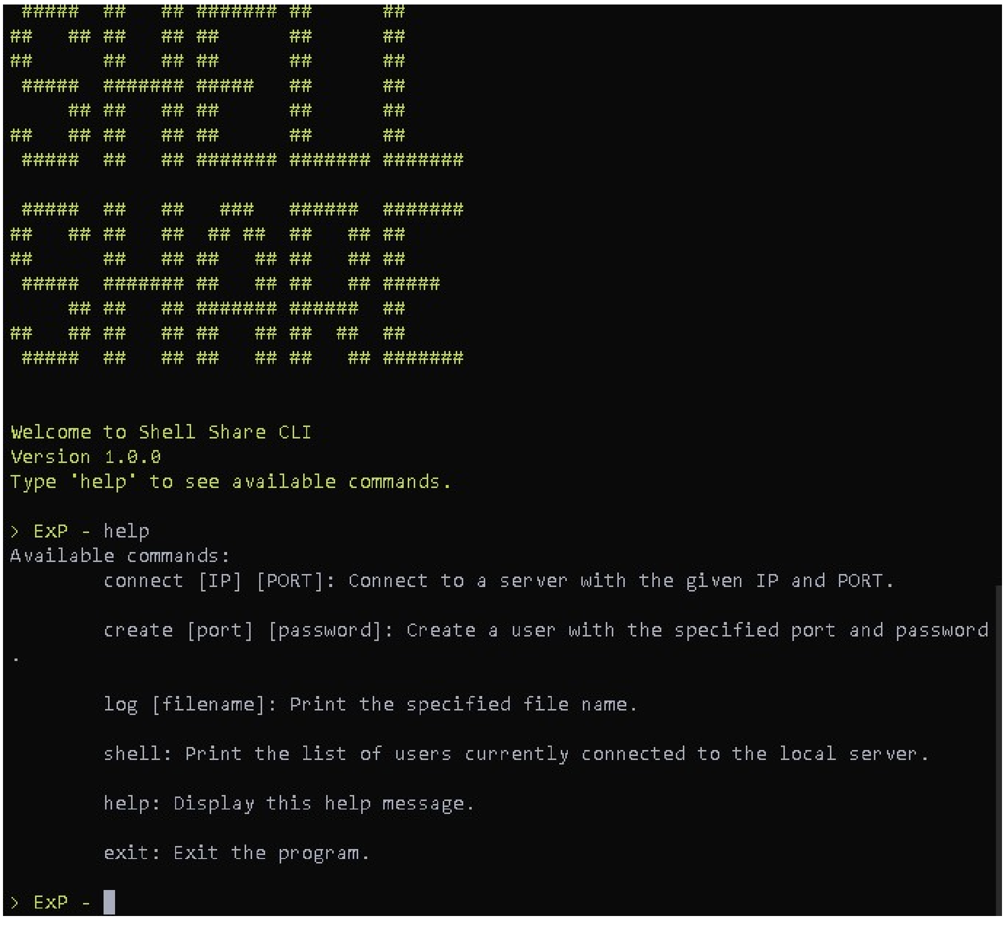
3. **create(args)**

   - Description: Creates a user with a specified port and password.
   - Parameters:
     - args:A list containing the port and password for the user.
   - Actions:
     - Retrieves the local IP address.
     - Prints the port, password, and user's IP address.
     - Starts a custom server process in the background using multiprocessing.

4. **log(args)**

   - Description:Logs actions to a specified file (not fully implemented).
   - Parameters:
     - args: A list containing the filename for logging.

5. **shell(args)**
    - Description:Displays a list of connected users by reading data from a CSV file.
    - Parameters:
        - args: Unused (for compatibility with other functions).
6. **help()**
    - Description:Displays the available commands and their usage.
7. **rambo_exit_animation()**
    - Description: Creates an exit animation for a graceful program exit.
8. **main()**
    - Description: Main function that initializes the Shell Share CLI.
    - Actions:
        - Prints a welcome message.
        - Accepts user commands and executes the appropriate actions based on the input.



```
 #####  ##   ## ####### ##      ##
##   ## ##   ## ##      ##      ##
##      ##   ## ##      ##      ##
 #####  ####### #####   ##      ##
     ## ##   ## ##      ##      ##
##   ## ##   ## ##      ##      ##
 #####  ##   ## ####### ####### #######

 #####  ##   ##   ###   ######  #######
##   ## ##   ##  ## ##  ##   ## ##
##      ##   ## ##   ## ##   ## ##
 #####  ####### ##   ## ##   ## #####
     ## ##   ## ####### ######  ##
##   ## ##   ## ##   ## ##  ##  ##
 #####  ##   ## ##   ## ##   ## #######


Welcome to Shell Share CLI
Version 1.0.0
Type 'help' to see available commands.

> ExP - help
Available commands:
        connect [IP] [PORT]: Connect to a server with the given IP and PORT.

        create [port] [password]: Create a user with the specified port and password
.

        log [filename]: Print the specified file name.

        shell: Print the list of users currently connected to the local server.

        help: Display this help message.

        exit: Exit the program.

> ExP -
```

Figure 3.2: Command-Line Interface.

3.1.3 Custom Server Creation

The script provides functions to create a custom server and connect to it as a client. It uses the socket module to establish connections over a network. The server can be configured to bind to a specific IP address and port, and it requires a password for client authentication.

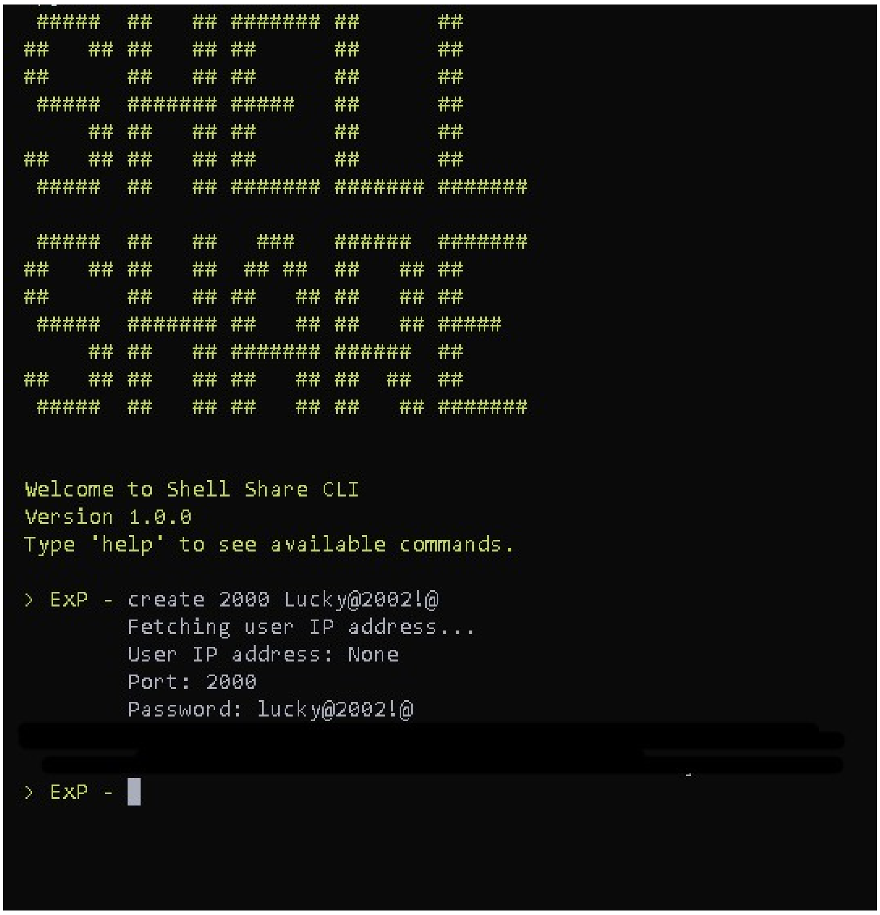
- **Features**

1. get_wifi_ip()
   - Description: Retrieves the local IP address based on a specific range.
   - Returns: The local IP address or None if not found.
2. start_custom_server(port, password)
   - Description: Starts a custom server that binds to a specified port and awaits client connections.
   - Parameters:
     - **port**: The port number to bind the server to.
     - **password**: The password required for client authentication.
   - Actions:
     - Binds the server to the specified port and awaits client connections.
     - Accepts clients, prompts for a nickname, and logs the client's information.
     - Authenticates clients using the provided password.
     - Handles client-server communication based on password validation.
3. connect_to_server(ip, port, password)
   - Description: Connects to a server with the specified IP address and port.
   - Parameters:
     - **ip**: The IP address of the server.
     - **port**: The port number to connect to.: The port number to connect to.
     - **password**: The password for server authentication.
   - Actions:
     - Creates a client socket and connects to the specified server.

  - Sends the provided password to the server for authentication.
  - Handles server responses based on password validation.

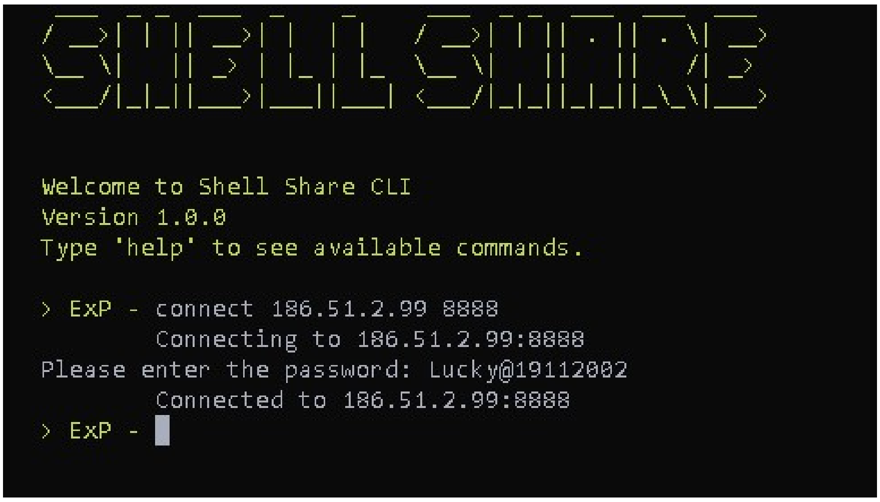
- **Usage**

To utilize the custom server creation functionalities, import the script into your project and utilize the start_custom_server and connect_to_server functions as needed.



```
 #####  ##   ## ####### ##      ##
##   ## ##   ## ##      ##      ##
##      ##   ## ##      ##      ##
 #####  ####### #####   ##      ##
     ## ##   ## ##      ##      ##
##   ## ##   ## ##      ##      ##
 #####  ##   ## ####### ####### #######

 #####  ##   ##   ###   ######  #######
##   ## ##   ##  ## ##  ##   ## ##
##      ##   ## ##   ## ##   ## ##
 #####  ####### ##   ## ##   ## #####
     ## ##   ## ####### ######  ##
##   ## ##   ## ##   ## ##  ##  ##
 #####  ##   ## ##   ## ##   ## #######


Welcome to Shell Share CLI
Version 1.0.0
Type 'help' to see available commands.

> ExP - create 2000 Lucky@2002!@
        Fetching user IP address...
        User IP address: None
        Port: 2000
        Password: lucky@2002!@
[redacted]

> ExP -
```

Figure 3.3: Create Commond .



```
 ___  _ _  ___  _    _     ___  _ _  ___  ___  ___
/ __>| | || __>| |  | |   / __>| | || . || . \| __>
\__ \|   || _> | |_ | |_  \__ \|   ||   ||   /| _>
<___/|_|_||___>|___||___| <___/|_|_||_|_||_\_\|___>


Welcome to Shell Share CLI
Version 1.0.0
Type 'help' to see available commands.

> ExP - connect 186.51.2.99 8888
        Connecting to 186.51.2.99:8888
Please enter the password: Lucky@19112002
        Connected to 186.51.2.99:8888
> ExP -
```

Figure 3.4: Connect Commond .

3.2 Project Flow

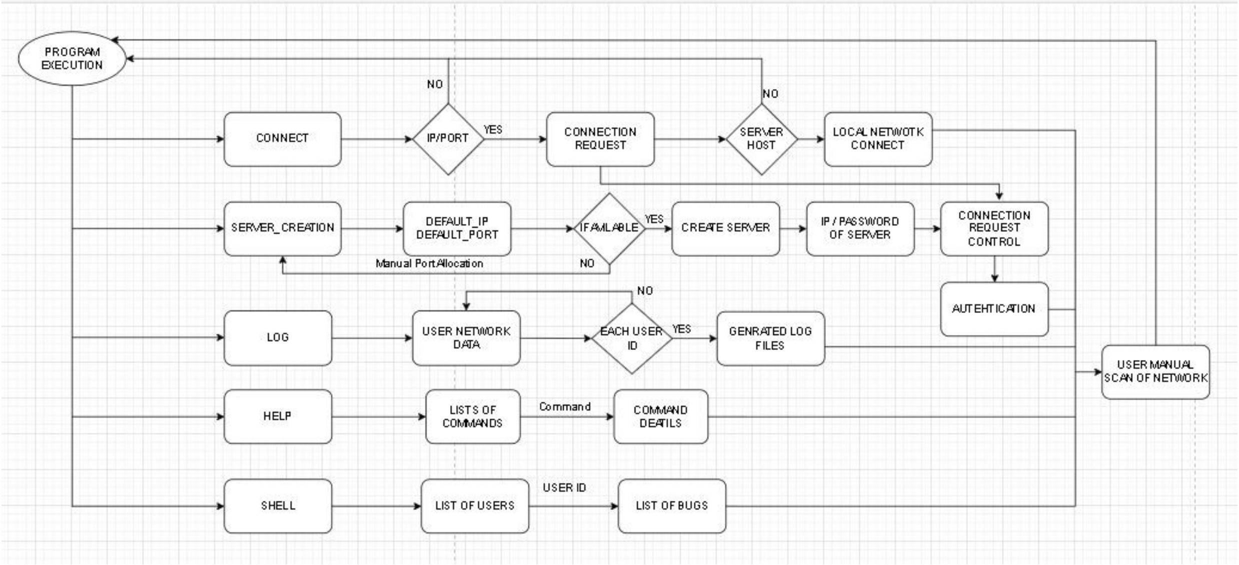


Figure 3.5: Project Flow Diagram.

3.3 TimeLine chart

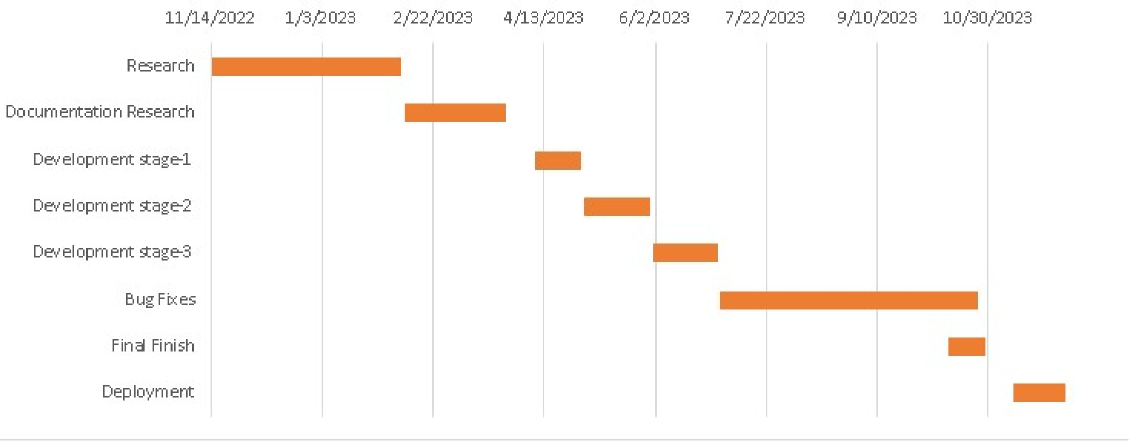


Figure 3.6: Time Line Chart.

Chapter 4

Future Work

- Create a LAN sub-network.
- Add 5+ users in one server.
- Implementation of GUI Interface

References

1. Brandon Rhodes, John Goerzen & Tim Bower, 2023, Foundations of Python Network Programming , 2rd ed., Prentice- APress.

2. Nathan Jennings, 2023 . Socket Programming in Python 1(1.2), pp. 0-30.

3. Pietro Grandest, 2022. Server deployment with Python: From A to Z , Prentice- Codementor.

4. Panagiotis Chartas, 2022. Villain Tool (Updated 12 January 2023) Available at: https://github.com/t3l3machus/Villain [Accessed 30 January 2023].

5. Python Documentation , 2022 . argparse — Parser for command-line options, arguments and sub-commands (3.2), pp. 2-30.

6. Adeyinka Ajao Adewale, Victor Olugbemiga Matthews, Adelakun Adebiyi, Olaitan Alashiri, 2021,packet-sniffer-for-users-end-network-performance-monitoring-using-python programming,, pp. 0-25.

7. R. B. Manjula, P. I. Basarkod, Yaser Mowlaiwzadah, 2021, Analyzing and Simplifying Log Files using Python, Vol. 9 , Prentice- IJERT.

8. Asswini Swain, Het Sheth, Pratik kanani, 2021, Remote Method Invocation Using Python, Vol. 6 , Prentice- International Journal Of Advanced Studies In Computer Science And Engineering,Ijascse.s

9. Adeyinka A. Adewale, Victor O. Matthews, Adebiyi A. Adelakun, Winifred Amase,Olaitan Alashir, 2019, Packet Sniffer for Users End Network Performance Monitoring using Python Programming, Prentice- International Journal Of Advanced Studies In Computer Science And Engineering,Ijascse.

10. V.Neethidevan, G.Chandrasekaran, 2019, Web Automation using Selenium Web driver Python , Prentice- International Journal of Recent Technology and Engineering.

11. Aditi Shrivastava, Nitin Shukla , 2019, Extracting Knowledge From User Access Logs , Prentice- International Journal of Scientific and Research Publications.

12. Hataw Jalal Mohammed , Kamaran Hama Ali Faraj, 2019, Python-WSGI and PHP-Apache Web Server Performance Analysis by Search Page Generator, Prentice- University of KURDISTAN Hewler.Hataw Jalal Mohammed , Kamaran Hama Ali Faraj, 2019, Python-WSGI and PHP-Apache Web Server Performance Analysis by Search Page Generator, Prentice- University of KURDISTAN Hewler.

13. Xia Qing, 2018, Network Security Management Platform System Design and Implementation, Vol. 7, Prentice- International Conference on Computer Engineering and 41 of 41 Technology.

14. Liu Zhijun, 2017, Computer network security virtual experiment system design and implementation, Prentice- International Conference on Intelligent Computation Technology and Automation.

15. HUANG Zhikuntt, 2017, The Design and Implementation of Security Network System Based on Web, Prentice- IEEE Workshop on Advanced Research and Technology in Industry Applications.

16. Dabin Sun, Bowei Wang, 2017, Research on the Design of the Implementation Plan of Network Security Level Protection of Information Security, Prentice- International Symposium on Mechatronics and Industrial Informatics.

Links:

https://docs.python.org/3/howto/argparse.html

https://docs.python.org/3/library/http.server.html

https://docs.python.org/3/library/argparse.html#argumentparser-objects

https:
//www.codementor.io/@pietrograndinetti/server-deployment-with-python-from-a-toz-1fjhy96qni

https://github.com/t3l3machus/Villain

https://realpython.com/python-sockets/

https://www.sourcecodester.com/python

https://realpython.com/python-sockets/#conclusion

https://docs.python.org/3/library/socket.html

https://www.researchgate.net/publication/341271544_Remote_Method_Invocation_Using_Python

https://chat.openai.com/