VISVESVARAYA TECHNOLOGICAL UNIVERSITY
BELAGAVI

Mini Project Report on

"NETWORK FIREWALL"

Submitted in the partial fulfillment for the requirements of Computer Graphics & Visualization Laboratory of 6th semester CSE requirement in the form of the Mini Project work

Submitted By

KAVYA S	USN: 1BY20CS082
LIKITHA D	USN: 1BY20CS097
MALAVIKA S PATIL	USN: 1BY20CS102

Under the guidance of

Mr. SHANKAR R
Assistant Professor, CSE, BMSIT&M

Prof. CHETHANA C
Assistant Professor, CSE, BMSIT&M

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BMS INSTITUTE OF TECHNOLOGY & MANAGEMENT
YELAHANKA, BENGALURU - 560064.

2022-2023

BMS INSTITUTE OF TECHNOLOGY &MANAGEMENT

YELAHANKA, BENGALURU – 560064

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the Project work entitled **"NETWORK FIREWALL"** is a bonafide work carried out by **Kavya S (1BY20CS082) , Likitha D (1BY20CS097) and Malavika S Patil (1BY20CS102)** in partial fulfillment for *Mini Project* during the year 2022-2023. It is hereby certified that this project covers the concepts of *Computer Graphics & Visualization*. It is also certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in this report.

Signature of the Guide with date	**Signature of the Guide with date**	**Signature of HOD with date**
Mr. Shankar R Assistant Professor CSE, BMSIT&M	Prof. Chethana C Assistant Professor CSE, BMSIT&M	Dr. Thippeswamy G Prof & Head CSE, BMSIT&M

EXTERNAL VIVA – VOCE

Name of the Examiners	**Signature with Date**
1. _____	_____
2._____	_____

INSTITUTE VISION

To emerge as one of the finest technical institutions of higher learning, to develop engineering professionals who are technically competent, ethical and environment friendly for betterment of the society.

INSTITUTE MISSION

Accomplish stimulating learning environment through high quality academic instruction, innovation and industry-institute interface.

DEPARTMENT VISION

To develop technical professionals acquainted with recent trends and technologies of computer science to serve as valuable resource for the nation/society.

DEPARTMENT MISSION

Facilitating and exposing the students to various learning opportunities through dedicated academic teaching, guidance and monitoring.

PROGRAM EDUCATIONAL OBJECTIVES

1. Lead a successful career by designing, analysing and solving various problems in the field of Computer Science & Engineering.
2. Pursue higher studies for enduring edification.
3. Exhibit professional and team building attitude along with effective communication.
4. Identify and provide solutions for sustainable environmental development.

ACKNOWLEDGEMENT

We are happy to present this project after completing it successfully. This project would not have been possible without the guidance, assistance and suggestions of many individuals. We would like to express our deep sense of gratitude and indebtedness to each and every onewho has helped us make this project a success.

We heartily thank our Principal, Dr. MOHAN BABU G N, BMS Institute of Technology &Management, for his constant encouragement and inspiration in taking up this project.

We heartily thank our Professor and Head of the Department, Dr. THIPPESWAMY G, Department of Computer Science and Engineering, BMS Institute of Technology &Management, for his constant encouragement and inspiration in taking up this project.

We gracefully thank our Project Guide, Mr. Shankar R, Assistant Professor and Prof. Chethana C, Assistant Professor, Department of Computer Science and Engineering for their intangible support and for being constant backbone for our project.

Special thanks to all the staff members of Computer Science Department for their help and kind co-operation.

Lastly, we thank our parents and friends for the support and encouragement given throughout in completing this precious work successfully.

KAVYA S (1BY20CS082)

LIKITHA D (1BY20CS097)

MALAVIKA S PATIL (1BY20CS102)

ABSTRACT

A **firewall** can either be software-based or hardware-based and is used to help keep a network secure. Its primary objective is to control the incoming and outgoing network traffic by analyzing the data packets and determining whether it should be allowed through or not, based on a predetermined rule set.

A network's firewall builds a bridge between the internal network or computer it protects, upon securing that the other network is secure and trusted, usually an external (inter)network.

Main aim of this Mini Project is to illustrate the concepts and usage of Fire Wall in OpenGL. We have used input devices like mouse and key board to interact with program. We have also used SolidCube for forming a complete network setup which help to understand concept of Congestion Control very well. To differentiate between objects we have used different colors for different objects. We have added menu which makes the program more interactive. We have used font family for indicating the name of objects as we can see in this project.

In this project we have used a small SolidCube to represent a data, which travels as data transfer from source to destination.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 BRIEF INTRODUCTION

As a software interface for graphics hardware, OpenGL's main purpose is to render two- and three-dimensional objects into a frame buffer. These objects are described as sequences of vertices or pixels. OpenGL performs several processing steps on this data to convert it to pixels to for final desired image in the frame buffer.

Computer graphics plays a crucial role in visually representing complex concepts and data structures, aiding in better understanding and analysis. The concept of the firewall was introduced to secure the communication process between various networks. A firewall is a software or a hardware device that examines the data from several networks and then either permits it or blocks it to communicate with your network and this process is governed by aset of predefined security guidelines.

OpenGL, a platform-independent and efficient graphics library, provides a powerful interfacefor rendering 2D and 3D graphics. It allows us to harness the graphical capabilities of modern computers to create dynamic and interactive visualizations. It allows us to harness the graphical capabilities of modern computers to create dynamic and interactive visualizations. A firewall is a device or a combination of systems that supervises the flow of traffic between distinctive parts of the network. A firewall is used to guard the network against nasty people and prohibit their actions at predefined boundary levels.

A firewall is not only used to protect the system from exterior threats but the threat can be internal as well. Therefore we need protection at each level of the hierarchy of networking systems. A good firewall should be sufficient enough to deal with both internal and external threats and be able to deal with malicious software such as worms from acquiring access to the network. It also provisions your system to stop forwarding unlawful data to another system.

OpenGL Fundamentals

This section explains some of the concepts inherent in OpenGL.

Primitives and Commands

OpenGL draws primitives—points, line segments, or polygons—subject to several selectable modes.

You can control modes independently of each other; that is, setting one mode doesn't affect whether other modes are set .Primitives are specified, modes are set, and other OpenGL operations are described by issuing commands in the form of function calls.

Primitives are defined by a group of one or more vertices. A vertex defines a point, an endpoint of a line, or a corner of a polygon where two edges meet. Data is associated with a vertex, and each vertex and its associated data are processed independently, in order, and in the same way. The type of clipping depends on which primitive the group of vertices represents.

Commands are always processed in the order in which they are received, although there may be an indeterminate delay before a command takes effect. This means that each primitive is drawn completely before any subsequent command takes effect. It also means that state- querying commands return data that's consistent with complete execution of all previously issued OpenGL commands.

1.2 MOTIVATION

The motivation for undertaking this project lies in addressing the inherent complexities of building network firewall technically, and the desire to enhance the learning experience for students and enthusiasts. Traditional methods of studying these visualization often rely on textual descriptions and static diagrams, which may not adequately convey the dynamic nature of their execution. This project seeks to leverage the power of computer graphics and the OpenGL framework to provide an immersive and interactive learning environment that enables users to visualize these important concepts of networking in real-time.

The visual representation of network firewall offers several significant advantages. First and foremost, it provides a more intuitive understanding of the inner structure, main components and working process of network firewall. By presenting the networking firewall simulation

process as a dynamic animation, users can observe how different protocols are relied in the building of firewall application. A firewall always exists between a private network and the Internet which is a public network thus filters packets coming in and out. Selecting a precise firewall is critical in building up a secure networking system .Firewall provisions the securityapparatus for allowing and restricting traffic, authentication, address translation, and content security. It ensures 365 *24*7 protection of the network from hackers. It is a one-time investment for any organization and only needs timely updates to function properly. By deploying a firewall there is no need for any panic in case of network attacks.

In conclusion, the motivation behind this project is to harness the power of computer graphicsand OpenGL to create an immersive and interactive learning experience working of network firewall. By providing real-time visualizations, this project aims to offer an intuitive and engaging environment where learners can grasp the intricacies of these algorithms. The visual representations not only enhance comprehension but also foster motivation and bridge the gap between theory and practice, ultimately equipping learners with a solid foundation in algorithmic thinking and problem-solving.

1.3 SCOPE

The scope of this project involves implementing and working of network firewall concept in Computer Networks using the OpenGL framework. It includes developing a graphical user interface (GUI) that allows users to interact with different underlying layers in networking and various protocols, modify input parameters, and observe the real-time working of the network firewalls and security provided by firewall. The project focuses on selecting a precise firewall is critical in building up a secure networking system .Firewall provisions the security apparatus for allowing and restricting traffic, authentication, address translation, and content security. The project also includes evaluating the effectiveness of the visualizations and the usability of the system.

1.4 PROBLEM STATEMENT

The problem addressed by this project is the difficulty in comprehending and understanding network firewall, which is fundamental component of Computer Networking in computer

science. Traditional methods of learning these mechanism, such as textual descriptions and static diagrams, may not effectively convey their dynamic nature and inner workings. As a result, students and enthusiasts face challenges in grasping the concepts, identifying patterns, and optimizing the working mechanism for efficient execution.

The lack of interactive and visual learning resources for Network firewall further exacerbates the problem. Without the ability to observe the step-by-step execution and visualize the overall working processes, learners may struggle to develop a deep understanding of the mechanism and their underlying protocols and mechanisms. This limits their ability to apply the effectively in practical scenarios and hinders their overall learning experience.

Therefore, the problem statement for this project is to develop a solution that utilizes the power of computer graphics and the OpenGL framework to provide interactive visual representations of working mechanism of Network firewall. The solution should aim to bridge the gap between abstract working concepts and their concrete implementation, offering learners a more intuitive and engaging learning experience. By leveraging OpenGL's graphical capabilities, the solution should enable users to interact with the networking protocols, modify input parameters, and observe the real-time execution, ultimately enhancing their comprehension and proficiency in working of Networking Firewall.

The visualizations leverage OpenGL's graphical capabilities to highlight key algorithmic steps, patterns, and optimizations, enhancing the understanding and comprehension of the algorithms

1.5 PROPOSED SYSTEM

The proposed system is an interactive computer graphics project that utilizes the OpenGL framework to create a visually engaging learning environment for working of networking firewall. It consists of a graphical user interface (GUI) that allows users to interact with the, networking layers, protocols, customize input parameters, and observe real-time visualizations. The system incorporates popular concept in Computer networking firewall. A network's firewall builds a bridge between the internal network or computer it protects, upon securing that the other network is secure and trusted, usually an external (inter)network

Main aim of this Mini Project is to illustrate the concepts and usage of Fire Wall in OpenGL and to show how a firewall works. A firewall can either be software-based or hardware-basedand is used to help keep a network secure. Its primary objective is to control the incoming and outgoing network traffic by analyzing the data packets and determining whether it shouldbe allowed through or not, based on a predetermined rule set.
A network's firewall builds a bridge between the internal network or computer it protects,upon securing that the other network is secure and trusted, usually an external (inter)network We have used input devices like the mouse and keyboard to interact with the program.

1.6 LIMITATIONS

Network firewalls are primarily designed to monitor and control network traffic based on predefined security rules. While they play a crucial role in network security, they are not directly involved in computer graphics processing. As such, it is not accurate to discuss the limitations of network firewalls in the context of computer graphics. Instead, let's focus on the limitations of computer graphics systems themselves. Here are some limitations you may encounter in computer graphics:

Memory Constraints: Computer graphics applications often require a significant amount of memory to store and manipulate data, including textures, geometry, and shaders. Insufficient memory can lead to performance issues or prevent the rendering of large or complex scenes.

Bandwidth Limitations: Transmitting large amounts of graphical data over a network can be challenging due to bandwidth limitations. This can impact real-time streaming of graphics or remote rendering scenarios.

Display Limitations: The quality and fidelity of computer graphics are ultimately limited by the capabilities of the display hardware. Factors such as resolution, color accuracy, and refresh rate can impact the visual experience.

Compatibility Issues: Different computer graphics systems may have varying hardware configurations, software versions, or file format support. Ensuring compatibility across platforms and devices can be challenging, particularly when transferring or sharing graphical content.

CHAPTER 2

LITERATURE SURVEY

2.1 Introduction to computer graphics:

Perhaps the dominant characteristic of this new millennium is how computer and communication technologies have become dominant forces in our lives. The combinations of computers and networks and the complex human visual system, through computer graphics, has led to new ways of displaying information, seeing virtual worlds and communicating with people and machines. Computer graphics is concerned with all aspects of producing pictures or images using a computer. The field began humbly almost 50 years ago, with the display of few lines on Cathode Ray Tube(CRT). We can create images by the computer that are indistinguishable from photographs of real objects. We routinely train pilots with simulated airplanes, generating graphical displays of a virtual environment in real time. Feature-length movies made entirely by computer have been successful, both critically and financially. Massive multiplayer games can involve tens of thousands of concurrent participants. OpenGL, a graphics software system has become a widely accepted standard for developing graphics applications. Fortunately OpenGL is easy to learn and it possesses most of the characteristics of other popular graphics system.

2.2 Applications of Computer Graphics

The development of computer graphics has been driven both by the needs of the user community and by the advances in hardware and software. The applications of computer graphics are many and varied; it can be divided into four major areas:

1. Display of information.

2. Design.

3. Simulation and animation.

4. User interfaces.

2.3 OpenGL Command Syntax

As you might have observed from the simple program in the previous section, OpenGL commands use the prefix gl and initial capital letters for each word making up the command name (recall glClearColor(),for example.) Similarly, OpenGL defined constants begin with GL_use all capital letters, and use underscores(_) to separate words(like GL_COLOR_BUFFER_BIT)

2.4 OpenGL-related Libraries

OpenGL provides a powerful but primitive set of rendering commands, and all higher level drawing must be done in terms of these commands. Therefore, you might want to write your own library on top of OpenGL to simplify your programming tasks. Also, you might want to write some routines that allow an OpenGL program to work easily with your windowing system. In fact, several such libraries and routines have already been written to provide specialized features, as follows.

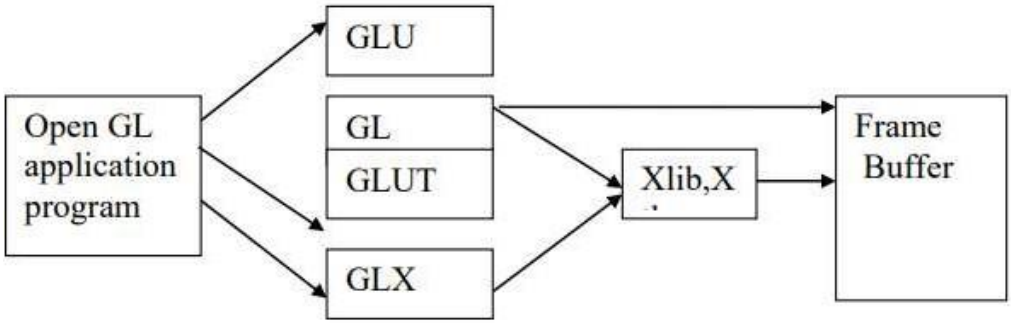

Figure 2.4 Block diagram of OpenGL Libraries

The OpenGL Utility Library(GLU) contain several routines that use lower-level OpenGL commands to perform such tasks as setting up matrices for specific viewing orientations and projections, performing polygon tessellation, and rendering surfaces. This library is provided as part of your OpenGL implementation. It's described in more detail in Appendix C and in the OpenGL Reference Manual. The more useful GLU routines are described in the chapters in this guide, where they're relevant to the topic being discussed. GLU routines use the prefixglu.

CHAPTER 3

SYSTEM REQUIREMENT

3.1 SOFTWARE REQUIREMENT

- Programming Language: Choose a language such as C++ or Java with strongOpenGL support.
- Integrated Development Environment (IDE): Utilize IDEs like Visual Studio, Code: Blocks, Eclipse, or NetBeans.
- OpenGL Library: Access OpenGL through libraries like GLUT or free glut.
- Graphics Drivers: Install up-to-date graphics drivers for optimal performance.
- Operating System: Specify the target OS for deployment (Windows, macOS, Linux).
- Additional Libraries and Tools: Consider libraries for user input handling or GUI development (e.g., GLFW, SDL, Qt).

Ensure compatibility among software versions and regularly update the software stack forstability.

3.2 HARDWARE REQUIREMENT

- Processor: Modern multi-core processor (e.g., Intel Core i5 or AMD Ryzen 5).
- Memory (RAM): Minimum of 8 GB RAM for efficient execution.
- Graphics Card: Dedicated card with at least 2 GB VRAM and OpenGL support.
- Display: High-resolution monitor (Full HD or higher).
- Input Devices: Standard keyboard and mouse.
- Storage: Adequate storage space (preferably an SSD).

Specific hardware needs may vary based on visualization complexity and algorithm requirements.

3.3 OPERATING SYSTEM

- WINDOWS XP,VISTA

CHAPTER 4

SYSTEM ANALYSIS

System analysis is an essential phase in the development process of any software project, including the representation of sea breeze prediction algorithms using OpenGL. It involves a thorough examination of the requirements, constraints, and objectives of the project, as well as the identification of potential solutions and strategies to meet those goals. these are the keyaspects to be considered during the system analysis:

- Requirements Gathering: Identify functional and non-functional requirements, including desired features, performance expectations, user interactions, and constraints.
- Use Case Analysis: Identify user roles, their goals, and expected system behavior for different scenarios.
- Data Analysis: Identify input, intermediate, and output data requirements, considering formats, sizes, and data dependencies.
- System Design: Determine the system architecture, software components, and their interactions, focusing on separation of concerns and scalability.
- Performance Analysis: Evaluate performance requirements, optimize algorithms, and ensure efficient handling of large datasets or complex visualizations.
- User Interface Analysis: Assess usability, interface layout, interactivity, visualizations, and user feedback mechanisms.
- Risk Analysis: Identify potential risks, challenges, and mitigation strategies to ensure project success.

By conducting a thorough system analysis, we can gain a clear understanding of the project requirements, constraints, and objectives, which serves as a foundation for the subsequent phases of system design, implementation, and testing.

CHAPTER 5

IMPLEMENTATION

This program is implemented using various openGL functions which are shown below.

Various functions used in this program.

glutInit() : interaction between the windowing system and OPENGL is initiated

glutInitDisplayMode() : used when double buffering is required and depth information is required

glutCreateWindow() : this opens the OPENGL window and displays the title at top of the window

glutInitWindowSize() : specifies the size of the window

glutInitWindowPosition() : specifies the position of the window in screen co-ordinates

glutKeyboardFunc() : handles normal ascii symbols

glutSpecialFunc() : handles special keyboard keys

glutReshapeFunc() : sets up the callback function for reshaping the window

glutIdleFunc() : this handles the processing of the background

glutDisplayFunc() : this handles redrawing of the window

glutMainLoop() : this starts the main loop, it never returns

glViewport() : used to set up the viewport

glVertex3fv() : used to set up the points or vertices in three dimension

glColor3fv() : used to render color to faces

glFlush() : used to flush the pipeline

glutPostRedisplay() : used to trigger an automatic redrawal of the object

glMatrixMode() : used to set up the required mode of the matrix

glLoadIdentity() : used to load or initialize to the identity matrix

glTranslatef() : used to translate or move the rotation centre from one point to another in three dimensions

glRotatef() : used to rotate an object through a specified rotation angle

Throughout the implementation process, it is essential to ensure that the project aligns with the requirements identified during the system analysis phase. Regular testing, optimization, and feedback collection will help to deliver a successful and dymanic Network firewall using OpenGL.

CHAPTER 6

INTERPRETATION OF RESULTS

Fig 6.1 Window displaying the menu.

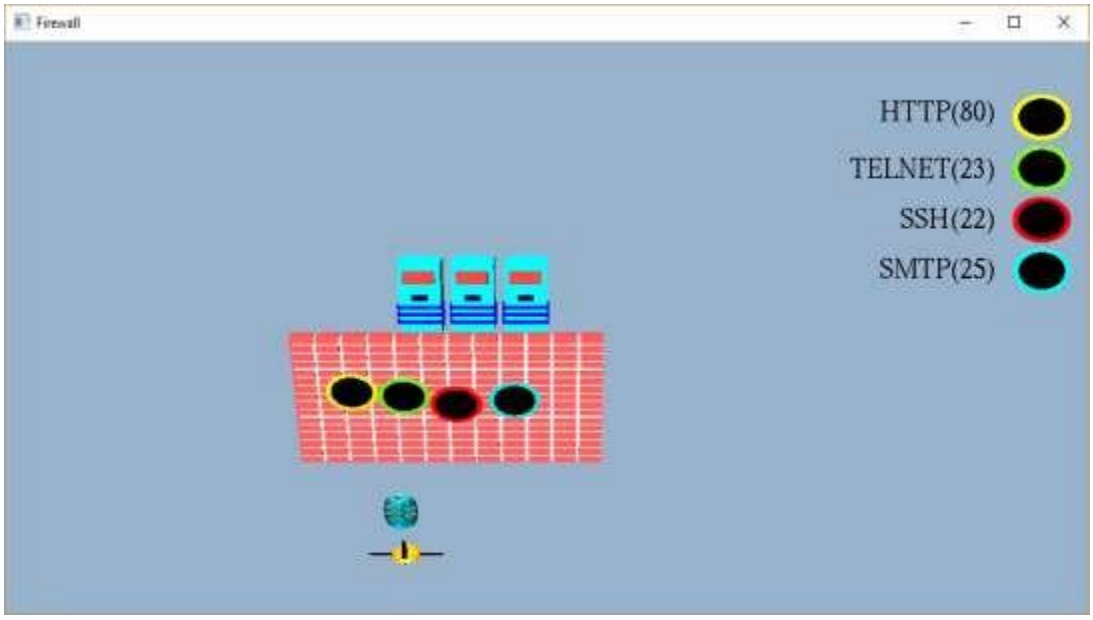

Fig 6.2 Window with external user, firewall, client-server and underlying protocols

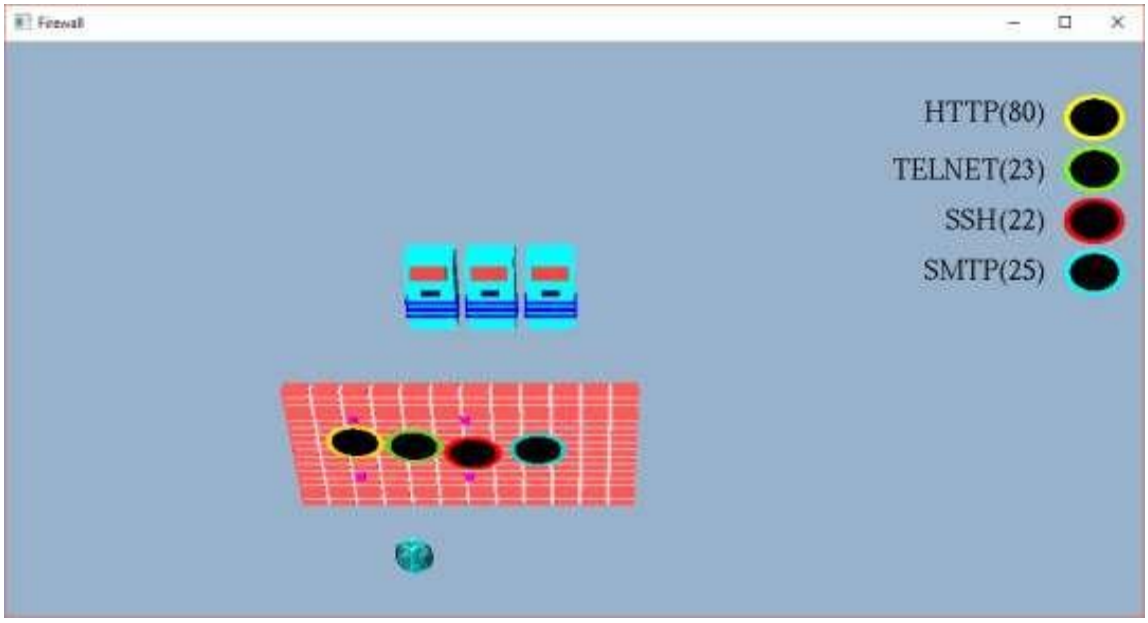

Fig 6.3 Window showing firewall for the network traffic within the private network.

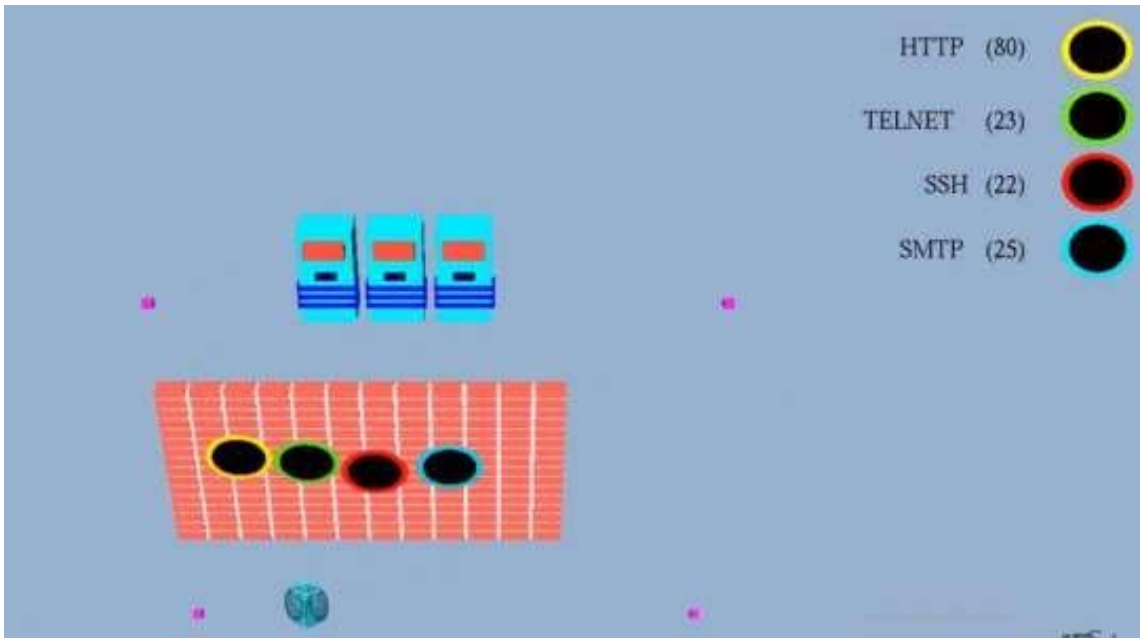

Fig 6.4 Dispatch of data packets after proper authorization

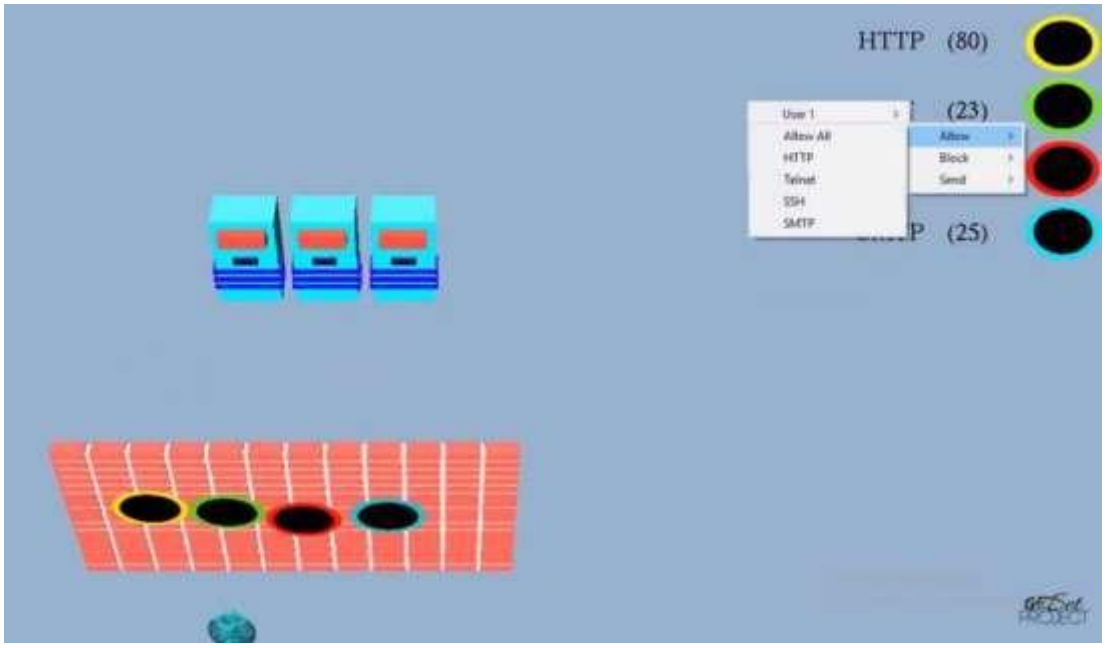

Fig 6.5 Customization of accepted protocols such as HTTP, Telnet, SMPT, SSH by dropdown to allow ,send and block specific protocols

CHAPTER 7

FUTURE ENHANCEMENT

A network firewall project could incorporate several future enhancements to improve its functionality and effectiveness.

Here are a few potential enhancements:

- **Deep Packet Inspection (DPI):** DPI involves analyzing the contents of network packets at a granular level, including inspecting the payload and application-layer data. By implementing DPI techniques within the firewall, it can identify and block specific types of network traffic, such as malicious payloads or unauthorized protocols. This enhances the firewall's ability to detect and prevent advanced threats.
- **Intrusion Detection and Prevention System (IDPS):** Integrating an IDPS into the firewall can provide an additional layer of security. The IDPS monitors network traffic for signs of suspicious or malicious activity, such as unauthorized access attempts or unusual data transfers.
- **Behavior-based Analysis:** Traditional firewalls often rely on signature-based detection methods, which can be less effective against emerging or unknown threats. Implementing behavior-based analysis techniques allows the firewall to learn normal network behavior patterns and identify anomalies that deviate from the expected behavior. This approach enables the firewall to proactively detect and respond to previously unseen threats.
- **Application-aware Firewall:** A firewall enhanced with application- awareness can understand the context and behavior of various network applications. This capability enables more granular control and allows the firewall to enforce application specific policies, such as restricting access to specific features or preventing data leakage through specific applications.
- **Cloud Integration:** As organizations increasingly adopt cloud services and infrastructure, integrating the firewall with cloud platforms can provide enhanced security. Cloud integration allows the firewall to leverage real-time threat intelligence and collaborate with other security components in the cloud

ecosystem. It can also enable centralized management and monitoring of firewall policies across distributed environments.

- **Machine Learning and Artificial Intelligence:** Leveraging machine learning and AI algorithms can significantly enhance the firewall's ability to detect and respond to evolving threats. By training the firewall with large datasets of network traffic and security information, it can learn to identify patterns and behaviors associated with known and unknown threats. This enables the firewall to make intelligent decisions and adapt its defense mechanisms in real-time.
- **Enhanced Visualization and Reporting:** In computer graphics, visualization plays a crucial role in conveying complex information effectively. Enhancing the firewall project with advanced visualization techniques can provide administrators with intuitive graphical representations of network traffic, security events, and system status. This allows for easier monitoring,analysis, and decision-making.

These are just a few potential enhancements for a network firewall project in the realm of computer graphics. It's important to note that implementing such enhancements would require expertise in network security, machine learning, and computer graphics to ensure effective integration and usability.

CHAPTER 8

8.1 CONCLUSION

The project "Network Firewall" is based on concepts of Network Security.

This program illustrates the concept of network firewall using various functions.

With potential future enhancements, such as Deep Packet Inspection, Intrusion Detection and Prevention System**,** Behavior-based Analysis, Application-aware Firewall**,** Cloud Integration and Machine Learning and Artificial Intelligence the project can become an even more comprehensive learning platform, contributing to computer graphics and enhancing mechanism exploration. Overall, this project has successfully demonstrated the value of OpenGL in creating informative and interactive visualizations for educational purposes in computer science and related disciplines.

Finally we conclude that this program clearly illustrate the functioning of a Firewall in OpenGL and has been completed successfully and is ready to be demonstrated.

8.2 REFERENCES

- **WEB REFERENCES**: http://jerome.jouvie.free.fr/OpenGl/Lessons/Lesson3.php
  http://google.com
  http://opengl.org www.wikipedia.com www.OpenGLsimpleexamples.com
  www.OpenGLprogramminguide.com
  https://www.youtube.com/watch?v=hwnZajb_A1chttp://www.freestudentprojects.com/
  https://getsetproject.com

- **TEXT BOOKS :**

  INTERACTIVE COMPUTER GRAPHICS A TOP-DOWN APPROACH

  -By Edward Angel.

  COMPUTER GRAPHICS,PRINCIPLES & PRACTICES :

  -Foley van dam

  -Feiner hughes