National University of Science and Technology
School of Electrical Engineering and Computer Science
Department of Computing

Final Year Project

Software Requirements Specifications

For

VPN Spyglass: VPN traffic analyzer

Version 2.0

Hassan Abdullah – 337275 – BSCS 10B
Sameen Mubashar – 346848 – BESE 11A

Advisor: Dr. Mehdi Hussain
Co – Advisor: Dr. Arsalan Ahmad

Dated: 20th September 2023

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Hassan	28th August, 2023	Requirements	1.1
Sameen	30th August, 2023	Interfaces Update	1.2
Sameen	2nd September, 2023	Non- Functional Requirements	1.3
Hassan	4th September, 2023	Overall Description Update	1.4
Hassan	13th September 2023	System Requirements	1.5
Sameen	18th September, 2023	Other Requirements and Glossary update	1.6
Hassan	20th September,2023	User interfaces Updated	2.0

1. Introduction

1.1 Purpose

The VPN SpyGlass project's goal is to provide a strong and user-friendly open-source tool for detecting and analyzing VPN activity within a network. This digital solution is designed to enable network managers to identify, categorize, and obtain insights regarding VPN activity on their networks. The program will distinguish between conventional network traffic and VPN traffic by utilizing cutting-edge deep packet inspection techniques, as well as categorize the observed VPN traffic depending on the exact protocol or type being utilized. The ultimate aim is to enable administrators to more effectively monitor network activity, identify unauthorized VPN connections, and minimize any security threats.

Users will have real-time insight over network traffic via a dynamic online dashboard powered by ReactJS / NextJS, Django, and Python, allowing them to take informed actions such as banning VPN connections using P4-based switches and OpenFlow rules or using PfSense Firewall. The VPN SpyGlass project seeks to bridge the gap between powerful network analysis and user-friendly interface design by providing a comprehensive solution for controlling VPN usage and improving network security.

1.2 Document Conventions

1.2.1 Document Conventions Used

This document uses the following conventions:

VPN	Virtual Private Network
BMV2	Behavioral Model Version 2
DPI	Deep Packet Inspection
GUI	Graphical User Interface
API	Application Programming Interface

Table 1: Document Conventions

1.2.2 Typographical Conventions

This document uses the following typographical conventions:

Font	Georgia
Italic	To draw attention to a particular point in text i.e., *Emphasis*
Underline	To enlist alternate words or to reference another part of text.
Bold	To highlight the **importance** of a Word/Phrase.

Table 2: Typographical Conventions

1.3 Intended Audience and Reading Specifications

The SRS is intended for programmers, project managers, testers, and reviewers of documentation.

The project, its goal, and its scope are introduced in the opening section of the text. The project is described in detail in the second section of the document, which also includes a walkthrough of the fundamental procedures to implement the functionalities, a description of the modes in which the application can be used, and functional and non-functional requirements and constraints.

- In order to ensure appropriate implementation, developers will use it as a fundamental guide to comprehend the specific functional and non-functional needs of the product.
- In order to coordinate their testing efforts with the anticipated results listed in the specification, testers will use the SRS to create thorough test cases.
- To track project progress, manage resource allocation, and make sure the development process complies with the specified criteria, project managers will consult the SRS.
- Users should have a fundamental awareness of the application's purpose and a quick grasp of how to utilize it.
- To understand the sequence of changes that have occurred, documentation writers should study earlier versions of the documentation.

1.4 Product Scope

The VPN SpyGlass system's goal is to provide advanced capabilities to network administrators for detecting, categorizing, and regulating VPN traffic within their network. The solution attempts to improve network security and visibility while also providing a user-friendly monitoring and management interface. The following are the product's key characteristics and expected outcomes:

Objectives:

1. **VPN Detection**: Use powerful packet analysis techniques to identify VPN traffic on your network.
2. **Categorization**: Create algorithms to classify VPN traffic depending on VPN protocol or kind, hence increasing visibility.
3. **Web Dashboard**: Create a user interface with real time traffic visualization and control options for a web dashboard.
4. **Real-time Data**: Integrate Mininet with the BMV2 switch to send real-time traffic data to the online dashboard.
5. **VPN blocking**: Enable administrators to restrict VPN traffic as needed by implementing OpenFlow rules.
6. **Performance**: Optimize system performance to reduce latency and resource usage.

Benefits:

1. **Enhanced Network Security**: The solution assists managers in identifying unauthorized VPN usage, allowing them to handle any security breaches as soon as possible.
2. **Monitoring User VPN behavior:** Administrators can monitor user VPN behavior to obtain insight into possible hazards and misuse.
3. **Granular Control**: The solution allows you to selectively prohibit VPN usage, giving you more control over network traffic.

1.5 Reference

[1] "Lightweight Anti DDoS Security Tool: Edge Level Filtering in SDN using P4," *IEEE Conference Publication | IEEE Xplore*, Feb. 22, 2023. https://ieeexplore.ieee.org/document/10179747/

[2] "Detection of VPN network traffic," *IEEE Conference Publication | IEEE Xplore*, Feb. 11, 2022. https://ieeexplore.ieee.org/document/9753621

[3] E. Papadogiannaki and S. Ioannidis, "A survey on encrypted network traffic analysis applications, techniques, and countermeasures," *ACM Computing Surveys*, vol. 54, no. 6, pp. 1–35, Jul. 2021, doi: 10.1145/3457904.

[4] G. Draper-Gil, A. H. Lashkari, M. S. I. Mamun, and A. A. Ghorbani, "Characterization of Encrypted and VPN Traffic using Time-related Features," *The International Conference on Information Systems Security and Privacy (ICISSP)*, Jan. 2016, doi: 10.5220/0005740704070414.

[5] IEEE Std. 830-1998, "IEEE Recommended Practice for Software Requirements Specifications, IEEE Computer Society," Software Engineering Standards Committee, 20 October 1998, No. SH94654. - References - Scientific Research Publishing. (n.d.). https://www.scirp.org/(S(vtj3fa45qm1ean45vvffcz55))/reference/References Papers.aspx?ReferenceID=358750

2. Overall Description

2.1 Product Perspective

The "VPN SpyGlass" project is a comprehensive network monitoring and management tool that detects and categorizes VPN activity in a network. It works in the context of network security and management, assisting network administrators in monitoring and regulating VPN usage.

2.1.1 System Architecture

The system follows a *client-server model*, consisting of two main components (Fig 1):

1. **Packet Analysis Engine**: This server-side component oversees identifying VPN traffic and doing deep packet inspection. Utilizing factors like IP addresses, port numbers, and packet durations, it analyses network packets and distinguishes between conventional and VPN traffic. Depending on the VPN protocol being utilized, detected VPN traffic is further categorized.
2. **Web Dashboard**: The client-side component offers administrators a graphical user interface for real-time monitoring and control and is accessed through web browsers. In order to get updates and command instructions, it connects to the server-side Packet Analysis Engine.

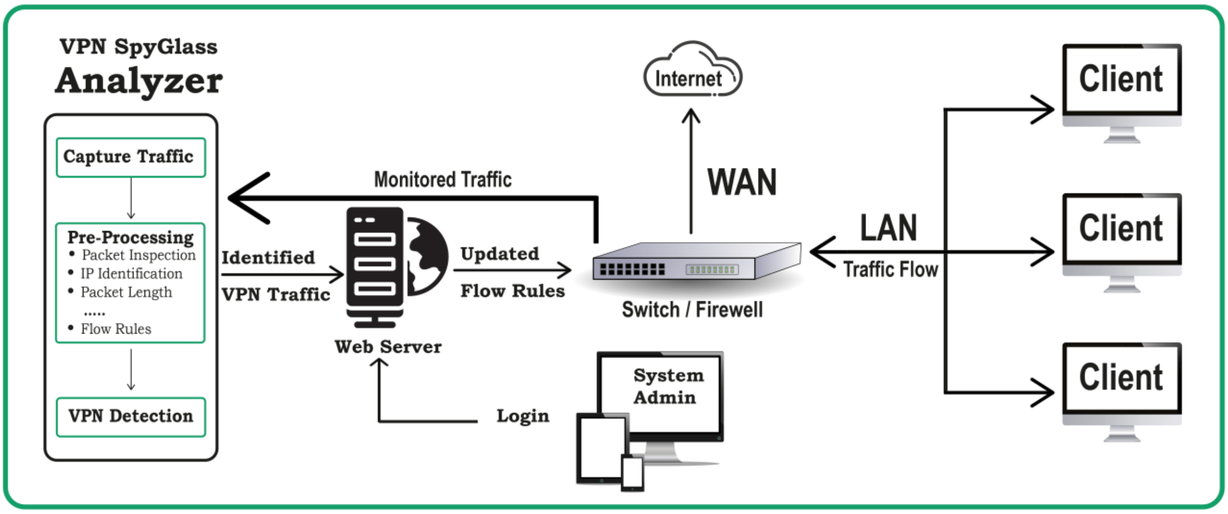


Fig 1: System Architecture

2.1.2 Interaction between Components

The incoming network packets are continually inspected by the Packet Analysis Engine. When VPN traffic is found, the Engine uses API calls to provide pertinent data to the Web Dashboard. This data is visualized via the Web Dashboard, which also shows real-time traffic changes and highlights VPN activity that has been discovered. Administrators may also start processes using the Dashboard, such as using OpenFlow/Firewall rules to block VPN traffic.

2.2 Product Functions

Our product automates the following functions for VPN traffic detection and monitoring:

2.2.1 Packet Capture and Inspection

- Capture network packets using Wireshark for analysis.
- Perform deep packet inspection to extract relevant packet attributes.

2.2.2 VPN Traffic Identifications

- Analyze packet attributes including IP addresses, port numbers, and packet lengths.
- Detect and identify VPN traffic based on predefined patterns.

2.2.3 VPN Protocol Classification

- Further classify detected VPN traffic based on the specific VPN protocol or type being used.

2.2.4 VPN Traffic Blocking

- Allow authorized administrators to block specific VPN traffic.
- Utilize OpenFlow/Firewall rules to implement traffic blocking on demand.

2.2.5 User Authentication and Access Control

- Implement user authentication to control access to the web dashboard.
- Ensure that only authorized users can view and control the system.

2.2.6 Graphical Representation of Insights

- Present data analytics and VPN traffic insights graphically in the web interface.
- Provide visualizations such as charts and graphs to represent network traffic trends.

2.2.7 Real-time Update of Insights

- Update graphical representations and visualizations dynamically as new VPN traffic is detected.
- Ensure that administrators have access to the most recent data and insights.

2.3 User Classes and Characteristics

2.3.1 Network Administrator

Responsible for managing and securing the network infrastructure. Have knowledge of network protocols, security measures, and traffic analysis techniques. Proficient in using technical tools for network monitoring and analysis.

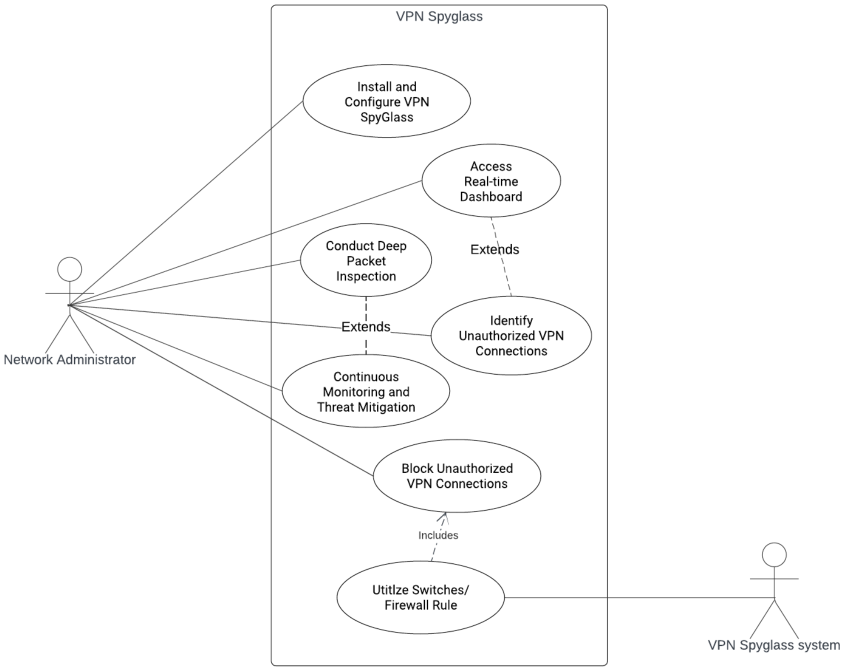

VPN Spyglass
Install and Configure VPN SpyGlass
Access Real-time Dashboard
Extends
Conduct Deep Packet Inspection
Identify Unauthorized VPN Connections
Extends
Network Administrator
Continuous Monitoring and Threat Mitigation
Block Unauthorized VPN Connections
Includes
Utitlze Switches/ Firewall Rule
VPN Spyglass system

Fig 2: Use Case Diagram

Use the VPN SpyGlass dashboard to monitor and analyze network traffic in real-time (Fig 2). Configure settings and rules to detect and categorize VPN traffic accurately. Utilize the dashboard to identify unauthorized VPN connections and potential security risks. Have the ability to block specific VPN traffic if required.

2.3.2 Software Developers

Program developers are responsible for designing, developing, and maintaining the VPN SpyGlass application. They possess strong programming skills and expertise in relevant technologies. Developers ensure the system's functionality, reliability, and adherence to design specifications.

Developers design and implement the deep packet inspection algorithm and the packet analysis engine responsible for VPN traffic detection and categorization. They create the web dashboard using ReactJS / NextJS, Django, and Python to provide a user-friendly interface for administrators. Developers integrate the different backend components, such as the Mininet topology, BMV2 switch, P4 language, and OpenFlow rules or using PfSense Firewall, to ensure real-time data transfer and control. They rigorously test the application's accuracy, performance, and security, identifying and resolving any issues.

2.4 Operating Environment

The program is an online application that can be accessed with any web browser that supports the **Secure Hyper Text Transfer Protocol (HTTPs).** The program will be installed on a cloud server. The website will be hosted **by Hostinger**. The client-side web app will be accessible through web browser on all devices. All operating systems, including Windows, MAC OS, Linux, and Android, will use the same version of the program.

Basic Browser- Latest Compatible Versions are as:

Google Chrome -110	*Opera - 92*
Mozilla Firefox - 106	*Microsoft Edge – 103*

Table 3: Compatibility

The backend implementation involves use of following:

Linux Ubuntu 22.04	Bazel – 6.0.0-pre.20220421.3	ONOS 2.20
Mininet 2.3.0	BMV2 (for behavioral model version 2)	PfSense

Table 4: Backend

2.5 Design and Implementation Constraints

The design and implementation constraints will include hardware, software, and security-usage constraints.

2.5.1 Timing Constraints

The overall size of all webpages should not exceed 5 Megabytes and the load time of a website should be kept under 3 seconds. so that page loads quickly on 97% of client devices.

Moreover, ensure that the tool's detection and analysis processes don't introduce significant delays, especially for real-time monitoring purposes. The web dashboard should provide real-time updates and responses, especially when administrators interact with it to block VPN traffic.

2.5.2 Design Constraints

The development process will be constrained by a number of implementation restrictions. The web dashboard should be designed to be responsive, ensuring that it works and displays properly on various screen sizes and devices, including desktops, tablets, and smartphones.

The user interface should be intuitive and user-friendly to facilitate easy navigation and interaction for network administrators. Maintain a consistent design and user experience across all sections of the web dashboard to avoid confusion. To promote readability and minimize congestion in the physical design, font sizes are fixed between 18 and 30.

2.5.3 Programming Constraint

To ensure that the website is accessible on all devices and is always updated, and the language doesn’t get outdated too soon, therefore, the website is designed using JavaScript, HTML, and CSS. **The React JS library** is one of the most updated programming libraries that can be used as React is a cross platform and platform independent language.

Moreover, P4 code should be written clean and well-documented to ensure that the project remains maintainable and understandable as it evolves. The code should be optimized for efficiency, especially when dealing with packet analysis and real-time monitoring to prevent performance bottlenecks.

2.5.4 Memory Constraints

Cookies will be stored for each individual user, storing login information as well as the user's consultation history. If the user has not been active in the past three months, the cookies will be erased.

For using PfSense firewall, the system will need minimum of 1 GB ram and 2 GB of Hard Drive to keep the system running.

Furthermore, space is necessary on the system to construct the virtual version of the BMV2 switch in Mininet topology. A minimum of 50 GB of space is necessary to keep the system running.

2.5.5 Hardware Constraints

The project requires a P4 programmable switch with Intel Tofino 2 chip. However, it is not available in the country and costs a lot therefore, virtual environment will be preferred instead of actual hardware.

Moreover, we can use firewalls like PfSense to implement it. As software based firewall, it can also be implemented using a PC or Laptop solely used for this purpose.

2.6 User Documentation

While launching the product, the user will be provided with guidelines which will help system administrators to use the product in the most efficient way possible.

2.6.1 Video Tutorial

Upon logging in or signing up for the VPN Spyglass website, you'll have access to a comprehensive video tutorial that covers all aspects of the tool. This tutorial aims to provide step-by-step guidance on how to use VPN Spyglass, ensuring that you can quickly grasp its functionality and features.

The video tutorial will cover the following topics:

- Introduction to VPN Spyglass
- Navigating the Dashboard
- Packet Analysis and Detection
- Real-time Monitoring
- Blocking VPN Traffic
- Using the Question Sections
- Troubleshooting Tips

2.6.2 FAQs

To further support users, VPN Spyglass features a dedicated section where you can find solutions to the most frequently asked questions. If you're facing a common issue or looking for specific information, the Question Sections can provide you with quick answers and solutions.

Some of the common topics covered in the Question Sections include:

- Getting Started with VPN Spyglass
- Troubleshooting and Error Resolution
- Understanding Packet Analysis and Detection
- Using Real-time Monitoring and Traffic Blocking
- Tips for Effective Network Management

By referring to the Question Sections, you can quickly find answers to queries that others have encountered, saving you time, and ensuring a smooth experience while using VPN Spyglass.

2.7 Assumptions and Dependencies

2.7.1 Dependencies

The dependencies include the following:

- **Packet Capturing Tool**: The system depends on a packet capturing tool like Wireshark or a similar utility to collect network traffic data.
- **Operating System Compatibility**: The tool's packet capturing component may have dependencies on specific operating systems (e.g., Windows, Linux) or versions of those operating systems.
- **Web Server**: The web dashboard component relies on a web server (e.g., Apache, Nginx) to host and serve the dashboard application.
- **Real-time Data Processing**: Real-time data processing and forwarding to the web dashboard may depend on a messaging system (e.g., WebSocket) or event handling framework.

- **Specific Versions of Software**: The product requires the system to have specific versions [1] of software to work properly.

2.7.2 Assumptions

The assumptions include the following:

- **User's Network Configuration**: It's assumed that the user has the necessary network permissions to capture and analyze network traffic, as well as to make changes to network rules if blocking VPN traffic is implemented.
- **Web Browser Familiarity**: Users are assumed to have a basic familiarity with using web browsers, including navigation, accessing web pages, and interacting with web elements.
- **Networking Knowledge**: Users are assumed to have a fundamental understanding of networking concepts, such as IP addresses, ports, and protocols, to interpret the information presented by VPN Spyglass accurately.
- **Web Dashboard Interaction**: Users are assumed to know how to interact with the web dashboard, including using its features, buttons, and controls effectively.
- **Internet Connectivity**: Users are assumed to have an internet connection to access the web dashboard and receive updates or alerts from VPN Spyglass.

3. External Interface Requirements

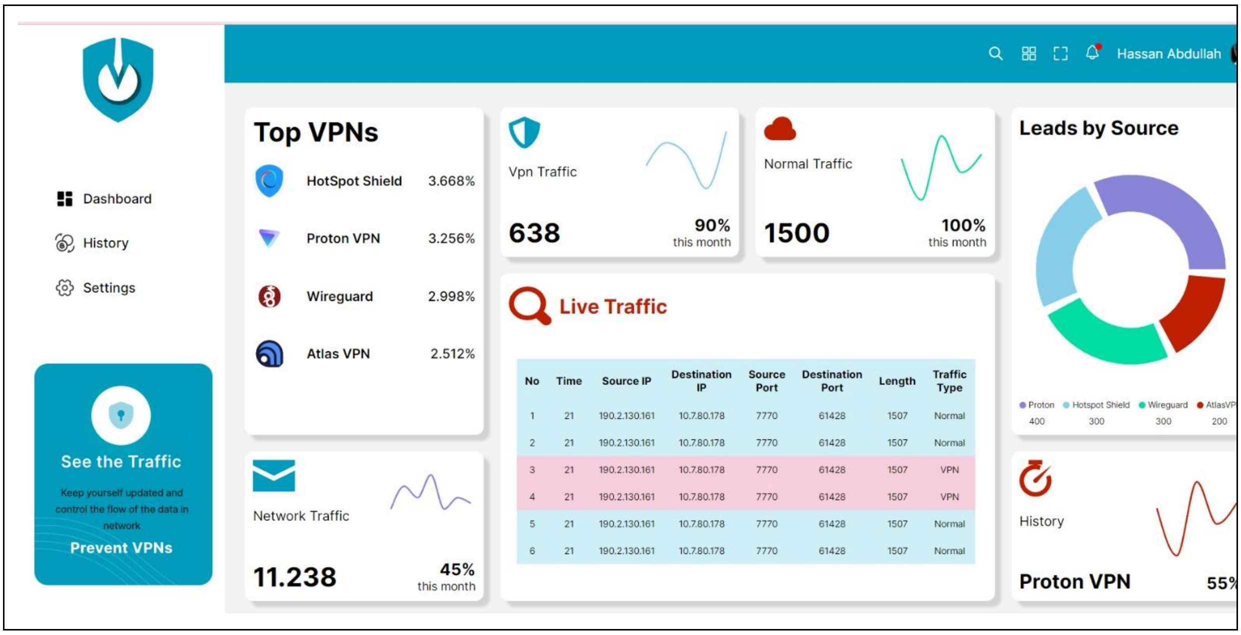
3.1 User Interfaces

The user interface for the VPN SpyGlass software shall consist of a web dashboard accessible through modern web browsers, including but not limited to Internet Explorer, Mozilla Firefox, Google Chrome, and Safari. The interface shall provide the following functionalities:

- Live Traffic Monitoring – Display real-time network traffic with color-coded indicators for regular and VPN traffic. Highlight and differentiate detected VPN traffic for easy identification.
- VPN Traffic Details – Show detailed information about identified VPN traffic, including IP addresses, port numbers, and protocols.
- VPN Blocking – Provide an option to block specific VPN traffic. Allow administrators to define OpenFlow/Firewall rules for traffic blocking.
- User Authentication – Require user authentication through a secure login process before granting access to the dashboard.
- Dashboard Configuration – Allow administrators to customize the dashboard layout, views, and preferences.

The user interface shall be developed using ReactJS / NextJS for front-end interactivity and Django for backend interactions.

[1] 2.4 Operating Environment - backend implementation



Keep yourself updated and control the flow of the data in network

Live Traffic

No	Time	Source IP	Destination IP	Source Port	Destination Port	Length	Traffic Type
1	21	190.2.130.161	10.7.80.178	7770	61428	1507	Normal
2	21	190.2.130.161	10.7.80.178	7770	61428	1507	Normal
3	21	190.2.130.161	10.7.80.178	7770	61428	1507	VPN
4	21	190.2.130.161	10.7.80.178	7770	61428	1507	VPN
5	21	190.2.130.161	10.7.80.178	7770	61428	1507	Normal
6	21	190.2.130.161	10.7.80.178	7770	61428	1507	Normal

Fig 3: GUI VPN SpyGlass

3.2 Hardware Interfaces

- **Network Hardware**: The system requires network devices (routers, switches, etc.) to capture and analyze network traffic.
- **Server Hardware**: A dedicated server or cloud infrastructure is needed to host the web dashboard and backend components.
- **Client Devices**: Users will access the web dashboard using devices such as laptops, desktops, tablets, and smartphones.

3.3 Software Interfaces

- **Operating System**: The system is compatible with multiple operating systems including Windows, macOS, and Linux.
- **Web Browsers**: The web dashboard supports modern web browsers such as Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge.
- **Wireshark**: The packet analysis engine utilizes Wireshark for capturing and inspecting network packets.
- **ReactJS / NextJS**: The frontend of the web dashboard is built using the ReactJS / NextJS.
- **Django**: The backend of the web dashboard is developed using the Django framework.
- **Python**: The packet analysis engine and backend components are developed using the Python programming language.
- **P4 Language**: The BMV2 switch uses the P4 language for defining packet processing behaviors.
- **PfSense**: Software based Firewall used to control the flow of traffic inside a network.

3.4 Communications Interfaces

- **HTTP/HTTPS**: Communication between clients and the web dashboard occurs over HTTP/HTTPS protocols.
- **API Endpoints**: The web dashboard communicates with backend components using RESTful API endpoints.
- **Wi-Fi/Ethernet**: The system captures network traffic through Wi-Fi or Ethernet interfaces depending on the network setup.
- **SSH**: To communicate between PfSense and the web dashboard, SSH commands and port communication will be used.
- **OpenFlow Protocol**: Communication between the web dashboard and the BMV2 switch occurs using the OpenFlow protocol for controlling the switch's behavior.
- **Interactive Forms**: The web dashboard uses interactive forms to display real-time traffic updates, VPN detection results, and user controls for VPN blocking.

4. System Features

4.1 Network Administrator

4.1.1 User Authentication

Description: Network administrators can log in to the VPN SpyGlass dashboard using their credentials.

Priority: High

Stimulus/Response Sequences:

- The network administrator accesses the VPN SpyGlass dashboard.
- The system prompts the administrator to enter their login credentials.
- Upon successful authentication, the administrator gains access to the dashboard.

4.1.2 Real-Time Traffic Monitoring

Description: Network administrators can monitor real-time network traffic using the dashboard.

Priority: High

Stimulus/Response Sequences:

- The network administrator logs in to the dashboard.
- The dashboard displays real-time traffic visualization.
- VPN traffic is highlighted for easy identification.

4.1.3 VPN Traffic Categorization

Description: The system automatically identifies and categorizes VPN traffic based on deep packet inspection.

Priority: High

Stimulus/Response Sequences:

- The network administrator observes the dashboard.
- VPN traffic is categorized and labeled by the system based on analysis results.

4.1.4 Unauthorized VPN Detection

Description: The dashboard helps network administrators identify unauthorized VPN connections.

Priority: High

Stimulus/Response Sequences:

- The network administrator reviews the dashboard.
- Unauthorized VPN connections are flagged, and alerts are generated.

4.1.5 VPN Traffic Blocking

Description: Network administrators can block specific VPN traffic using OpenFlow rules through the dashboard.

Priority: High

Stimulus/Response Sequences:

- The network administrator accesses the dashboard.
- In cases of security concerns, the administrator selects specific VPN traffic to block.
- OpenFlow rules are applied to block the selected traffic.

Action	**Response**
User Authentication	• The network administrator accesses the VPN SpyGlass dashboard. • The system prompts the administrator to enter their login credentials. • Upon successful authentication, the administrator gains access to the dashboard.
Real-Time Traffic Monitoring	• The network administrator logs in to the dashboard. • The dashboard displays real-time traffic visualization. • VPN traffic is highlighted for easy identification.
VPN Traffic Categorization	• The network administrator observes the dashboard. • VPN traffic is categorized and labeled by the system based on analysis results.
Unauthorized VPN Detection	• The network administrator reviews the dashboard. • Unauthorized VPN connections are flagged, and alerts are generated.
VPN Traffic Blocking	• The network administrator accesses the dashboard. • In cases of security concerns, the administrator selects specific VPN traffic to block. • OpenFlow rules are applied to block the selected traffic.

Table 5: Network Administrator

4.2 Software Developers

4.2.1 DPI Algorithm Implementation

Description: Software developers design and implement the deep packet inspection (DPI) algorithm for identifying VPN traffic.

Priority: High

Stimulus/Response Sequences:

- Developers work on implementing the DPI algorithm.
- DPI algorithm analyzes network packets for VPN traffic patterns.

4.2.2 Web Dashboard Development

Description: Developers create the web dashboard using ReactJS with NextJS, Django, and Python.

Priority: High

Stimulus/Response Sequences:

- Developers design and develop user-friendly web dashboard.
- The dashboard provides real-time traffic monitoring and control features.

4.2.3 Backend Integration

Description: Developers integrate backend components, including the Mininet topology, BMV2 switch, P4 language, and OpenFlow rules, or using Firewall to enable real-time data transfer and control.

Priority: High

Stimulus/Response Sequences:

- Developers ensure seamless integration of backend components for data collection and control.
- OpenFlow / PfSense flow rules are implemented for traffic blocking.

4.2.4 System Testing

Description: Developers rigorously test the application for accuracy, performance, and security.

Priority: High
Stimulus/Response Sequences:

- Developers conduct thorough testing to identify and resolve any issues.
- Performance and security vulnerabilities are addressed during testing.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

- The system shall process and categorize incoming packets for VPN detection in real-time with minimal latency.
- The web dashboard shall load and display real-time traffic updates within 2 seconds.

- The deep packet inspection algorithm shall process a minimum of 1000 packets per second.
- The tool shall handle up to 500 simultaneous users accessing the web dashboard without experiencing performance degradation.
- The system should provide accurate VPN detection with a false positive rate of less than 5%.
- The system must be capable of satisfying 99/100.

5.2 Security Requirements

- User authentication for the web dashboard shall use strong encryption (e.g., HTTPS) to protect login credentials.
- The system shall not log or store any sensitive user data or network traffic data.
- Access to the web dashboard shall be protected against unauthorized access.
- The system shall comply with relevant data protection and privacy regulations.
- The system shall not introduce vulnerabilities into the network it is monitoring.

5.3 Safety Requirements

- The system shall not interfere with the normal operation of the network it is monitoring.
- In the event of a system failure or crash, it shall recover and resume normal operation within 1 minute.

5.4 Software Quality Attributes

- The user interface (UI) of the web dashboard shall be intuitive and user-friendly, with a maximum learning curve of 2 hours for new users.
- The system shall maintain logs for auditing purposes, storing them securely and ensuring they are only accessible to authorized personnel.
- The system should provide regular software updates and patches to address security vulnerabilities and improve functionality.
- The web dashboard shall be responsive and compatible with major web browsers (e.g., Chrome, Firefox, Safari).
- The system shall be designed to be scalable, allowing for easy expansion to handle larger networks.

5.5 Business Rules

- The system shall not block VPN traffic without administrator authorization.
- Administrators must have the capability to customize and fine-tune VPN detection rules to accommodate specific network requirements.
- The system shall maintain an audit trail of VPN blocking actions for accountability.
- Only authorized administrators shall have access to the VPN blocking feature.
- The system shall not block VPNs used for legitimate business purposes without explicit user consent.

Appendix A: Glossary

Word	Definition
API	Application Programming Interface. A set of defined methods and functions that allow different software components to communicate with each other.
Authentication	The process of confirming the identity of a user, device, or system.
Backend	The part of a software system responsible for data processing, storage, and business logic.
BMV2 Switch	Behavioral Model Version 2 switch, used for software-defined networking research and development.
Compatibility	The ability of different systems or components to work together without issues.
Django	A high-level Python web framework for rapid development and clean, pragmatic design.
DPI	Deep Packet Inspection. A technique used for analyzing network traffic at the packet level to identify patterns, protocols, and behaviors.
Frontend	The user-facing part of a software system, responsible for presenting data and enabling user interaction.
GUI	Graphical User Interface. The visual interface that allows users to interact with the software.
IP Address	Internet Protocol address. A numerical label assigned to each device connected to a computer network.
Latency	The time delay between a user's action and the response of the system.
Live Traffic Monitoring	The functionality of the web dashboard that displays real-time network traffic information.
Mininet	An emulator that creates a realistic virtual network environment for testing, research, and learning.
Network Administrator	A person responsible for managing and maintaining a computer network.

Network Traffic	The flow of data packets within a computer network.
NextJS	A React framework for server rendered React applications.
OpenFlow Rules	Rules defined in the OpenFlow protocol that specify how network traffic should be handled by switches.
P4 Language	Programming language for specifying how packets are processed by network devices.
Packet Analysis Engine	The core component of VPN SpyGlass responsible for analyzing network packets to detect and categorize VPN traffic.
Packet Length	The size of a packet, usually measured in bytes.
Port Number	A numerical identifier for a specific endpoint of a communication session in a computer network.
Python	A widely used programming language known for its simplicity and versatility.
ReactJS	A JavaScript library for building user interfaces.
Real-Time	Events or actions that occur instantly or with minimal delay.
SRS	Software Requirements Specification. A formal document that outlines the requirements for the software project.
User Authentication	The process of verifying the identity of users before granting them access to the web dashboard.
VPN Blocker	A feature of the web dashboard that allows administrators to block VPN traffic within the network.
VPN Protocol	A specific set of rules and technologies used to establish and manage a Virtual Private Network.
VPN SpyGlass	The project being developed, which is a tool for detecting and categorizing VPN traffic within a network.
Vulnerabilities	Weaknesses or flaws in a system that could be exploited by malicious actors.

Web Dashboard	The user interface that provides real-time monitoring, control, and configuration features to administrators.
Wireshark	A popular open-source packet analyzer used for network troubleshooting, analysis, software development, and education.