Software Requirements Specification

for

Signature and Machine Learning based Web Application Firewall

Prepared by – Prachi Sharma

Manya Varshney

Priyansha Singhal

KIET Group Of Institutions

26th April 2023

Table of Contents

Table of Contents ii

Revision History ii

1. Introduction 1

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Product Scope 1

2. Overall Description 2

2.1 Product Perspective 2

2.2 Product Functions 2

2.3 User Classes and Characteristics 2

2.4 Operating Environment 3

2.5 Design and Implementation Constraints 3

2.6 Assumptions and Dependencies 3

3. External Interface Requirements 3

3.1 User Interfaces 3

3.2 Hardware Interfaces 3

3.3 Software Interfaces 3

3.4 Communications Interfaces 3

4. System Features 4

4.1 System Feature 1 4

4.2 System Feature 2 (and so on) 4

5. Other Nonfunctional Requirements 4

5.1 Performance Requirements 4

5.2 Safety Requirements 5

5.3 Security Requirements 5

5.4 Software Quality Attributes 5

5.5 Business Rules 5

6. Other Requirements 5

Appendix A: Glossary 5

Appendix B: Analysis Models 5

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |

# Introduction

## Purpose

An unprotected website is a security risk to customers, other businesses, and public/government sites. It allows for the spread and escalation of malware, attacks on other websites, and even attacks against national targets and infrastructure. In many of these attacks, hackers will try to harness the combined power of thousands of computers and sites to launch this attacks, and the attacks rarely lead directly back to the hackers.

A WAF protects your web apps by filtering, monitoring, and blocking any malicious HTTP/S traffic traveling to the web application, and prevents any unauthorized data from leaving the app. It does this by adhering to a set of policies that help determine what traffic is malicious and what traffic is safe.

The purpose of this project is to develop a web application firewall that uses Signature-Based Detection (SBD), Anomaly-Based Detection (ABD) and Artificial Neural Networks (ANN) as one of the artificial intelligence techniques.

## Document Conventions

The document focuses on the high priority requirements which will be implemented for the final deliverable

## Intended Audience and Reading Suggestions

*The document is intended for requirements engineer, domain expert, developer and project manager. Before reading this document it is highly recommended to read the general working of a firewall to get an overview of the product.*

## Product Scope

*The market for web application firewalls is expanding as a result of the growing relevance of web applications and the rise in web threats such cyber theft, espionage, vandalism, and fraud.*

*Additionally, during the projection period, market prospects for web application firewall are predicted due to the advent of web application firewalls driven by machine learning and artificial intelligence (AI) and increased penetration among SMEs.*

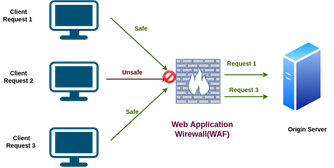
*A web application firewall's benefits include defense against automated temporary patching, zero-data exploits, and data leak prevention.*

# Overall Description

## Product Perspective

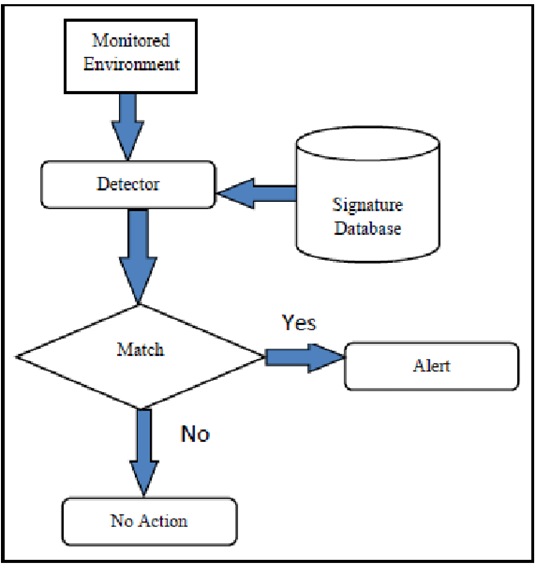
*This firewall uses signature based detection of the http requests that is integrated with the anomalous detection technique for detecting malicious requests sent to the server.*

*By using SBD, the detections were made against known web-based attack types such as SQL (Structured Query Language) Injection, Cross-Site Scripting (XSS), Command Injection, and Directory Traversal Attacks. HTTP requests that do not conform to the structure of the web application architecture were detected using ABD.*



## Product Functions

*This data flow diagram uses the detector to find and compare activity signatures found in the monitored environment to the known signatures in the signature database. If a match is found, an alert is issued and there is no match the detector does nothing.*



## User Classes and Characteristics

*The intended user is expected to be technically proficient in the basics of cybersecurity. The user having some basic background knowledge is expected.*

## Operating Environment

*This firewall can be used by these operating systems:*

* *Windows*
* *Linux*
* *AIX*
* *Solaris*
* *Mac OSX*

## Design and Implementation Constraints

*One major constraint for the Web Applications depends on how many people are connected. It may take time to fetch the rulesets from the database to get them configured if multiple people are doing it at the same time which may cause latency. Another factor is also internet connection because this is an always online feature. The application is getting data from the database over the internet and without it, it won’t function.*

## Assumptions and Dependencies

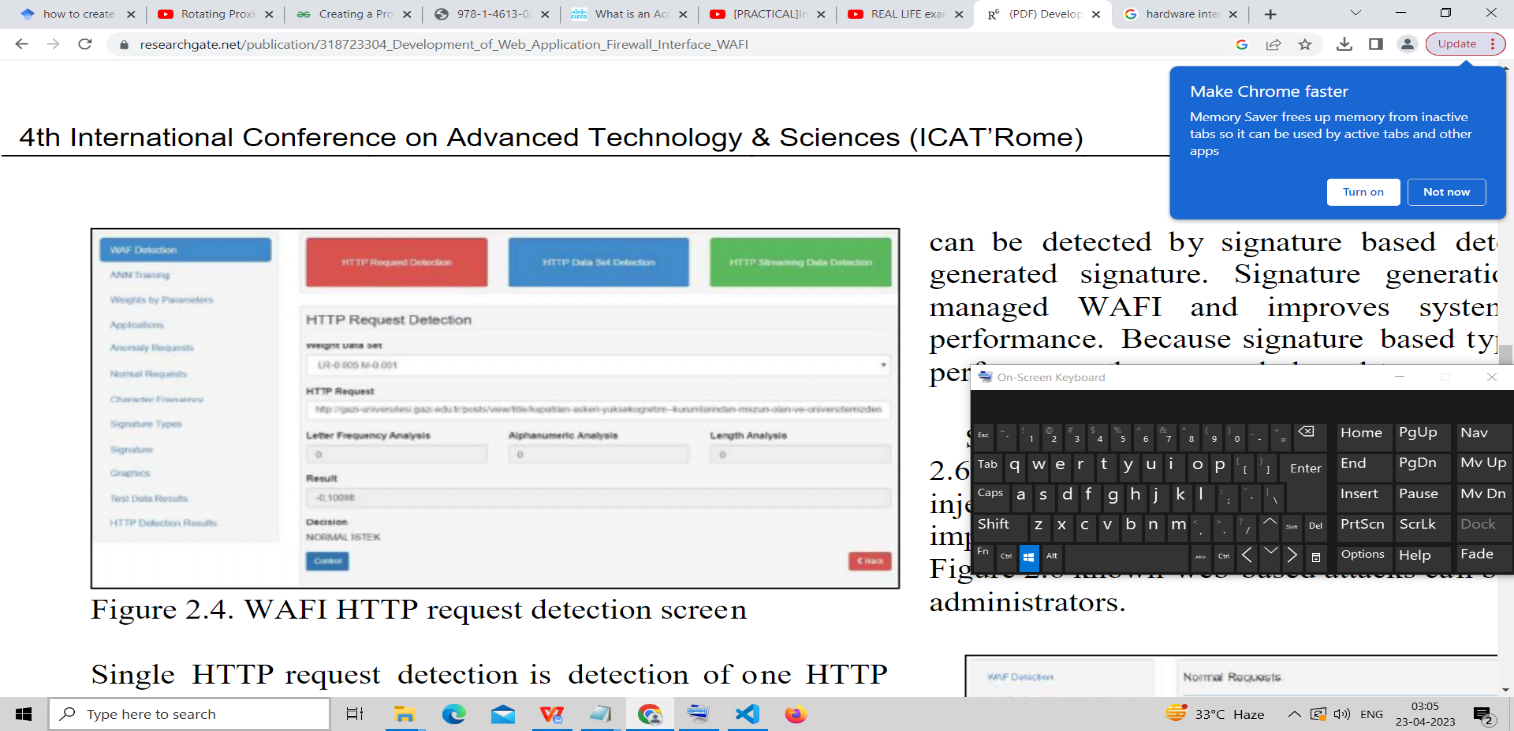
*It is assumed that the client is familiar with an internet browser and familiar with handling the keyboard and mouse. Since the application is a web-based application there is a need for the internet browser. It will be assumed that the users will possess decent internet connectivity. The application also requires the management of servers. As a result, it is assumed the client has their own server system or has access to an application*

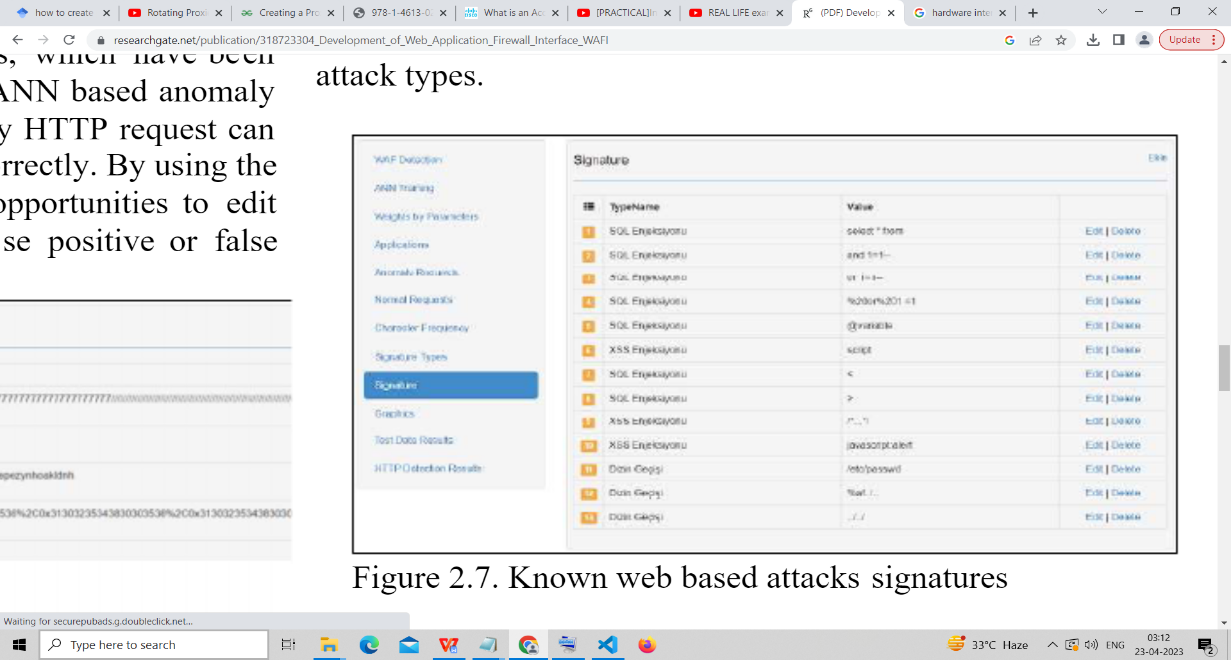
# External Interface Requirements

## User Interfaces

*WAFI (Web Application Firewall Interface) through which a user interacts with web application security. WAFI determines how commands are given to the WAF and how information is displayed on the screen. WAFI have three main types of visual interaction*

1. *Instructions: the user needs to know the WAF and web application security specific instructions.*
2. *Menus: user choose the commands from the menus, displayed on the screen and manage the attack detection of web application.*
3. *Graphical user interface (GUI): user gives commands by selecting and clicking on icons displayed on the screen. With the proposed WAFI, it seems to provide convenience to system administrators for both system setup and evaluation of results.*





## Hardware Interfaces

*This Software Product works at the Application Layer of OSI Model, Port 8080 and HTTP protocol .*

*The WAF is a transparent proxy in front of the application to ensure that all traffic passes through it and separately sends filtered traffic to the application, hiding the IP address of the application service. It essentially monitors and parses all requests based on the rules (called policies) before they reach your web server/ application. These policies equip the Web App Firewall to ensure that malicious requests and payload do not degrade, compromise, or expose your applications to DDoS or other threats, or cause an exfiltration of data. The policies tell the firewall what needs to be done if vulnerabilities or misconfigurations are found.*

## Software Interfaces

*This Software Product uses Python 3.10.7 and Visual Studio Code version 1.7 in the production of proxy server and Signature Based Detection. It uses Jupiter Notebook version 3.10 for Anomaly Based Detection. For Database, It is using MongoDB version 6. The Flow of HTTP Request is as follows:*

*Client --> Proxy Server --> Database (Test) --> Signature (Test) --> Anomaly (Test) --> Accept*

*or*

*Block*

## Communications Interfaces

The web browser is the client-side application that sends requests to the web server. The web server is the server-side application that receives requests from the web browser and responds with the requested information. The communication between the web browser and web server is based on the Hyper Text Transfer Protocol(HTTP).The WAF acts as Communication Interface Between Web Browser and Web Server.

# System Features

* 1. ***Your next firewall must scan for threats in allowed collaboration applications – e.g., Sharepoint, Box.net, MS Office Online***
     1. ***Description:*** *Enterprises continue to adopt collaborative applications hosted outside their physical locations. Whether it’s hosted Sharepoint, Box.net, Google Docs, or Microsoft Office Live, or even an extranet application hosted by a partner, many organizations have a requirement to use an application that shares files – in other words, is a high-risk threat vector.*

***4.1.2 Functional Requirements:*** *First step is to identify the application (regardless of port or encryption), allow it, and then scan it for any of the appropriate threats – exploits, viruses/malware, or spyware…or even confidential, regulated, or sensitive information.*

***4.2 The firewall makes network security simpler, not more complex with the addition of application control.***

* + 1. ***Description:*** *Many enterprises struggle with incorporating more information feeds and more policies, and more management into already overloaded security processes and people. In other words, if teams cannot manage what they’ve already got, adding more management, policies, and information doesn’t help.*
    2. ***Functional Requirements:*** *: Firewall policy should be based on user and application. Subsequent content analysis can be performed on allowed traffic, but fundamental access control should be based on relevant elements (i.e., application and user or group). This can have a significant simplifying effect. Firewall policy based on port and IP address, followed by subsequent analysis to understand the application makes things more complicated than they are today*

# Other Nonfunctional Requirements

## Performance Requirements

WAF implementation is the process of integrating a WAF into a web application or a web application delivery infrastructure to provide an additional layer of protection against various web-based attacks. The implementation typically involves configuring the WAF to enforce a set of security policies designed to detect and block malicious traffic targeting the web application. This may include defining rules to inspect incoming HTTP/S traffic for patterns and anomalies that indicate an attack, such as SQL injection or cross-site scripting (XSS).WAF implementation may also require integrating the WAF with other security solutions, such as intrusion prevention systems (IPS) and security information and event management (SIEM) platforms, to enable better threat detection and response capabilities. Overall, WAF implementation is a critical aspect of web application security that helps protect against a wide range of attacks that target web applications and web services.

## Security Requirements

*The Databases of Normal and anomalous HTTP Requests are needed to be secured using Identity Authentication System.*

## Software Quality Attributes

The WAF has the following Software Quality Attributes : adaptability, availability, flexibility, interoperability, maintainability, portability, reliability, reusability, robustness, testability, and usability.

## Business Rules

*Specify Access Permission details (not available if Path-specific routing is selected).*

***Allowed Client Networks***

*Select or add the allowed networks that should be able to connect to the hosted web server.*

***Blocked Client Networks***

*Select or add the denied networks that should be blocked to your hosted web server.*

***Authentication***

*Select a web app authentication profile or click Create new to create a new authentication profile. You can also create an authentication profile from :*

*Device Configuration > Protect > Web Server > Authentication Policies page.*

# Other Requirements

The following items provide a partial list of system attributes that can serve as requirements that should be objectively verified.

**6.1 Reliability**

The factors needed to establish the software’s required reliability.

**6.2 Availability**

Specify the factors needed to guarantee a defined level of availability.

**6.3 Security**

Specify the factors that will protect the software from accidental or malicious access, misuse, or modification.

These factors may include:

• cryptography

• activity logging

• restrictions on intermodule communications

• data integrity checks

**6.4 Maintainability**

These requirements may relate to modularity, complexity, or interface design. Requirements should not be placed here simply because they are thought to be good design practices.

**6.5 Portability**

Specify attributes of the software that relate to the ease of porting the software to other host machines and/or operating systems.

Appendix A: Glossary

Signature Based Detection - a style of detection that provides a mechanism for how the detector scans the data to find attacks

Hyper Text Transfer Protocol – HTTP is a protocol for fetching resources such as HTML documents

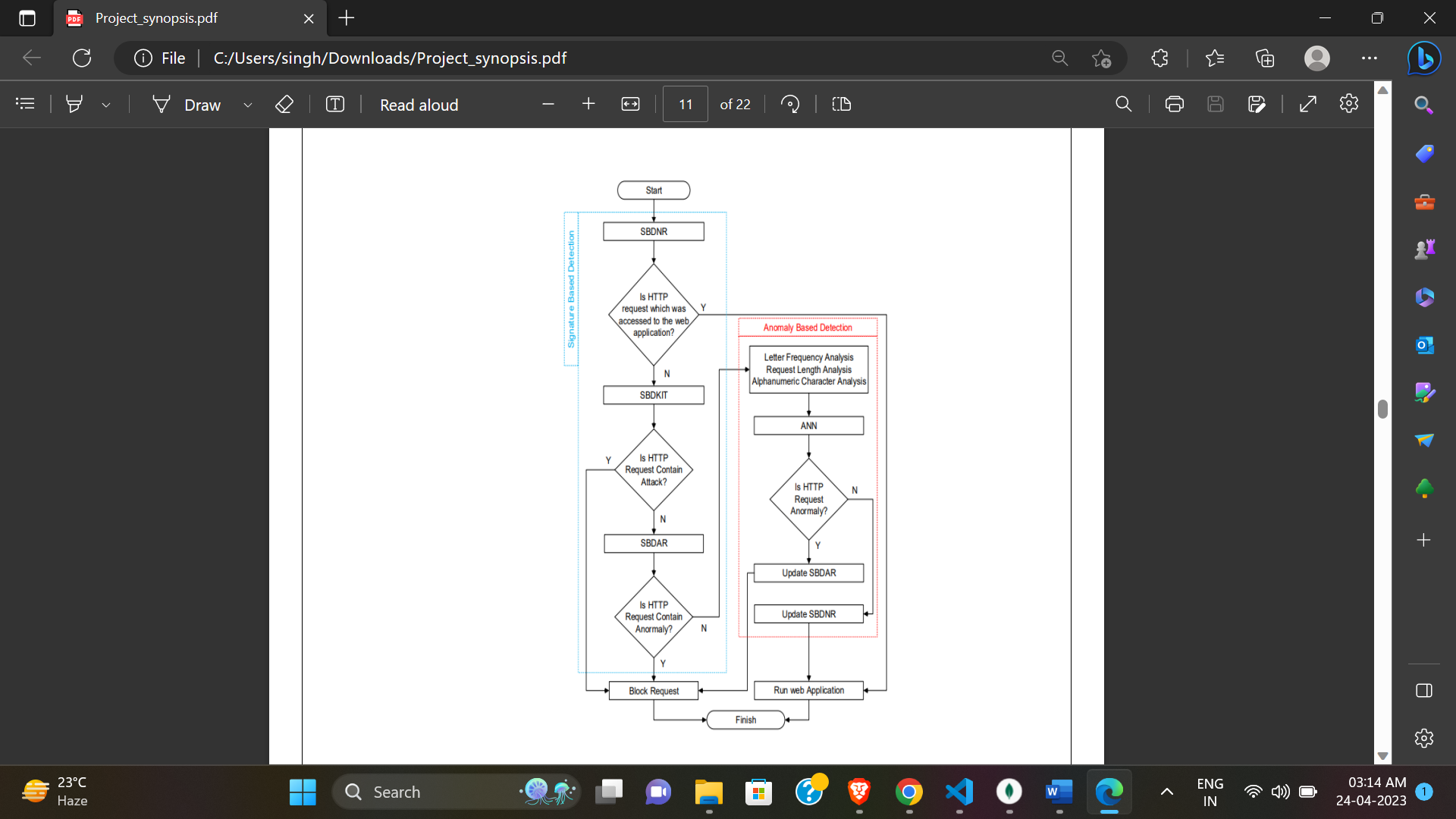
Open System Interconnection - The OSI model describes seven layers that computer systems use to communicate over a network

Anomaly Based Detection - an intrusion detection system for detecting both network and computer intrusions and misuse by monitoring system activity and classifying it as either normal or anomalous.

Artificial Neural Network - an interconnected group of nodes, inspired by a simplification of neurons in a brain

**Appendix B: Analysis Models**

**FlowChart**

****