System Requirements Specification

Team: Lighthouse Solutions

Version: 1.0

Date: 10/6/2022

Client: Ben Drozdenko of theNaval Undersea Warfare Center Division Newport (NUWCDIVNPT)

Prepared by Johnny Driscoll, Sean Staton, Dylan Haughton, Brody Looney and Andy Howe

Table of Contents

Abnormal Network Traffic Flow Dashboard Tool

System Requirements Specification

1. Introduction

In order to fulfill the University of Maine's undergrad requirements, the Lighthouse team has been chosen to develop software for a client. This software is for Ben Drozdenko representing the Naval Undersea Warfare Center Division Newport (NUWCDIVNPT). The project is a LAN Traffic Analyzer designed to flag repeated abnormal data that appears in the network.

1.1 Purpose of This Document

This document is available for a summary of the intentions and requirements of the software as well as the deliverables scheduled completion dates.

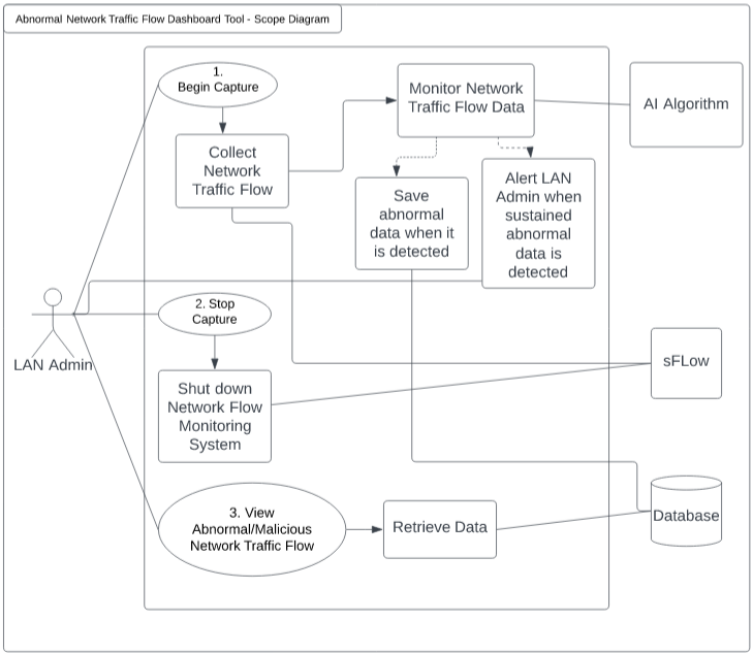
1.2. References

There are no external references related to this document.

1.3. Purpose of the Product

Purpose is to give the client a viable means of analyzing traffic that moves around their network in order to protect the integrity of confidentiality of all data that flows through.

1.4. Product Scope



This diagram above includes the three use cases detailed in the Functional Requirements section. The LAN admin can begin to capture live network traffic flow data, our tool will begin to collect this data a process it into the correct format using a network flow monitoring system. Our AI algorithm will then process the data and monitor it, constantly checking it for abnormalities. If sustained abnormal data is discovered by the algorithm the LAN admin will be alerted and the data will be saved to the database for further investigation. A user will be able to stop the capture, this will turn the network flow monitoring system off, the AI algorithm will not be running anymore since no data will be flowing into the machine. Lastly, the LAN admin will be able to view any abnormal data which is pulled from our database that stores the abnormal data. This data will be shown to the LAN admin in a list format with relevant information about the data.

2. Functional Requirements

Below are the major functional requirements of the project.

1. The system shall have a user interface.
2. The system shall have a user interface with no bloat elements.
3. The system shall allow the user to begin capturing network traffic flow data.
4. The system shall allow the user to stop capturing network traffic flow data.
5. The system shall collect network data.
6. The system shall discern which data are statistically abnormal to the user.
7. The system shall allow for the user to decide if data are abnormal or not.
8. The system shall alert the LAN admin when consistent, sustained abnormal activity has been observed
9. The system shall allow the user to view individual cases of abnormal network traffic flow
10. The system shall be able to be trained from a file at any point
11. The system shall be able to be trained from capture

Use Cases:

- This system has a very lightweight UI which is shown through these use cases. The system will allow for the user to do three simple things: they can begin a capture, stop the capture, and lastly view abnormal network traffic flow data. Other than these use cases, the system functions in an internal manner. The system will complete the other nine functional requirements above without any input from the user.

Number	1	
Name	Begin Capture	
Summary	The user will be presented an option to ‘begin capturing’, therein data will begin training the program live.	
Priority	1	
Preconditions	The application is open.	
Postconditions	System is recording data and training the program.	
Primary Actor	General users	
Secondary Actors	Target Network, Network Flow Monitoring System	
Trigger	Selecting the ‘begin capture’ button	
Main Scenario	**Step**	**Action**
	1. User opens the application.	System displays the UI.
	2. User hits the ‘begin capture’ button.	System begins recording and interpreting data, graph reflects this to the user
Extensions	**Step**	**Branching Action**
	2a	if alert is found: Display alert for abnormal data
Open Issues		

Number	2	
Name	Stop Capture	
Summary	If the user wants to stop capturing network traffic flow they can choose to stop the capture by selecting the ‘stop capture’ button	
Priority	1	
Preconditions	The capture is actively running	
Postconditions	The capture will be ended	
Primary Actor	General user	
Secondary Actors	Target Network, Network Flow Monitoring System	
Trigger	User hits the ‘stop capture’ button.	
Main Scenario	**Step**	**Action**
	1. User opens the application.	System displays the UI.
	2. User hits the ‘stop capture’ button.	System ends the recording and interpreting data, the graph will display previously captured data
Extensions	**Step**	**Branching Action**
Open Issues		

Number	3
Name	View Abnormal Network Traffic Flow
Summary	The user will be able to view abnormal network traffic flow
Priority	1

Preconditions	● User has the application downloaded onto their device. ● Abnormal network traffic flow has been detected	
Postconditions	System will store the abnormal network traffic flow for future use.	
Primary Actor	General user	
Secondary Actors	Database	
Trigger	Selecting the 'View Abnormal Network Traffic Flow' button	
Main Scenario	**Step**	**Action**
	1. User opens the application.	System displays the UI.
	2. User hits the 'View Abnormal Network Traffic Flow' button.	System opens the screen for Abnormal Network Traffic Flow and displays a list format of all the abnormal data
Extensions - None	**Step**	**Branching Action**
Open Issues		

Functional requirement tests

1. Test that the system starts collecting data when the capture button is pressed.
2. Test that the system analyzes live data.
3. Test that the system stops collecting data when the stop capture button is pressed.
4. Test that the system displays current network flow data to the UI.
5. Test that the system sends a report to the UI when abnormal activity is detected.
6. Test that the system sends a message to the LAN admin when sustained suspicious activity is detected.
7. Test that the system is able to be trained by a network traffic flow dataset
8. Test that the system is able to be trained by a capture of the network traffic flow dataset
9. Test that individual cases of abnormal network traffic flow data can be viewed.

3. Non-Functional Requirements

Number	NFR	Priority
1	The system shall detect abnormal activity 70% of the time	5
2	The system shall have a graph to display network data to the user	5
3	The system shall update the display information within 5 seconds of it changing	3
4	The system shall only incorrectly label normal packets as abnormal 5% of the time	5
5	The system shall be available to run 24 hours a day, 365 days a year, 80% of the time	5
6	The system shall detect abnormal activity within 5 seconds of its connection	4
7	The system shall be capable of handling 1000 connections a day	3
8	The system shall only be accessible by authorized personnel	3
9	The system shall integratable with hardware available to the naval ships	5
10	The system shall work properly when run on a machine with 64gb ram or greater	3

4. User Interface

See User Interface Design Document for *Abnormal Network Traffic Flow Dashboard Tool*.

5. Deliverables

Delivery Items	Delivery method	Delivery Date
SRS	Video Call /GitHub	October 13th (Tentative)
SDD	Video Call / GitHub	November 10th
UIDD	Video Call / GitHub	November 29th
CDRD	Video Call / GitHub	December 8th
CIR	Video Call / GitHub	TBA
AM	Video Call / GitHub	TBA
UG	Video Call / GitHub	TBA
FPR	Video Call / GitHub	TBA

6. Open Issues

- As a development team we need to conduct an investigation to determine what type of AI Algorithm we will plan on using to monitor the network traffic flow data.

Appendix A – Agreement Between Customer and Contractor

This document states the software requirements and specifications for the *Abnormal Network Traffic Flow Dashboard Tool* contracted by Ben Drozdenko and the Naval Undersea Warfare Center Division Newport (NUWCDIVNPT). Both the customer and the development team agree that this document clearly states the problem at hand, requirements, and deliverable deadlines for this project.

In the event that there are future changes made to this document the development team and customer will discuss the changes during our bi-weekly check-in meetings. If everyone agrees on the changes, we will document the change below and sign-off showing that the amendment has been approved.

Brody Looney	Brody Looney	10/19/2022
Typed Name	Signature	Date
Dylan Haughton	Dylan Haughton	10/19/2022
Typed Name	Signature	Date
Sean Staton	Sean Staton	10/19/2022
Typed Name	Signature	Date
Johnathan Driscoll	Johnathan Driscoll	10/20/2022
Typed Name	Signature	Date
Andrew Howe	Andrew Howe	10/20/2022
Typed Name	Signature	Date
Typed Name	Signature	Date

Customer Comments:

Appendix B – Team Review Sign-off

Andy Howe, Dylan Haughton, Johnny Driscoll, Sean Staton, and Brody Looney have all reviewed this document. We all have agreed on the content that listed in this document is accurate and complete. We all agree that the format follows the correct structure. The signatures below prove the information stated previously is correct.

Brody Looney	Brody Looney	10/19/2022
Typed Name	Signature	Date
Dylan Haughton	Dylan Haughton	10/19/2022
Typed Name	Signature	Date
Sean Staton	Sean Staton	10/19/2022
Typed Name	Signature	Date
Johnathan Driscoll	Johnathan Driscoll	10/20/2022
Typed Name	Signature	Date
Andrew Howe	Andrew Howe	10/20/2022
Typed Name	Signature	Date

Comments:

Appendix C – Document Contributions

Group Member	Sections Worked On	Contribution Percentage
Sean Staton	Functional Requirements, Use Cases, Revisions	20%
Dylan Haughton	Introduction	20%
Andy Howe	Deliverables, Functional Requirement Tests	20%
Johnny Driscoll	Non-functional Requirements	20%
Brody Looney	Product Scope, Functional Requirements, Appendix A, Appendix B, and Appendix C	20%