Vision Document

For an Intrusion Detection System using a Neural Network

Version 2.0

Submitted in partial fulfillment of the requirements of the degree of MSE

Blake Knedler

CIS 895 – MSE Project

Kansas State University

Table of Contents

1 Introduction

This document will provide the vision for an Intrusion Detection System (IDS) using a back propagation trained neural network. This IDS is a host-based system that will monitor network traffic aimed at detecting malicious intrusions. This Vision Document will describe the main functionality through Unified Modeling Language (UML) diagrams and requirements for an IDS that uses a neural network to determine if certain network traffic is acceptable or intended to be malicious.

1.1 Motivation

Malicious software developers are extremely common in our present time. Many of these developers aim to find weaknesses in others' networks to either exploit information or to cripple a network. Both of these goals can cause a great deal of harm and be extremely costly to those victims. Malicious software has become very sophisticated and difficult for simple virus protection applications to stop. It is very difficult to recognize these malicious attacks quickly and effectively. Newer techniques need to be used in order to gain an upper hand on the malicious software attacking many networks across the world. A simple IDS can help solve this problem.

The IDS watches for and detects when malicious intent is aimed at the network of the IDS. By listening on the host computer's network cards the system will be able to monitor all traffic that is received and sent from that device. The system will include a neural network that is trained on popular data through backpropagation in order to quickly detect these attacks. The neural network's speed is crucial for this project in order for the system to effectively detect the malicious intent and determine what actions to take. This system's aim is to detect and notify a user of a malicious attack, but could be expanded to prevent these attacks as well.

1.2 Terms and Definitions

1.2.1 IDS

An Intrusion Detection System (IDS) is a system that monitors network traffic for malicious attacks on the host system. The IDS will aim at determining the difference between normal network traffic and malicious traffic. Once the system notices malicious traffic it typically performs an action such as notifying a user or preventing any more traffic to the malicious host.

1.2.2 LAN

A local area network (LAN) is a computer network that is very limited in scope compare to a wide area network. A local area network is a

collection of computers in a small area such as a home. Groupings of LANs makeup a WAN.

1.2.3 WAN

A wide area network (WAN) is a computer network that is very broad and encompasses many users. In the case of the paper's topic, these networks typically all are contained within an internet service provider. A particular user sub-network on this WAN would be the network that is under attack.

1.2.4 DOS Attack

A denial of service (DOS) Attack is a type of network attack that aims to take down a network by flooding it with useless messages. These types of attacks are the most common type of network attack.

1.2.5 Artificial Neural Network

An artificial neural network (ANN) or commonly referred to as a neural network is a software design aimed at decision making using data. This system is designed after the neural network of our brains. A neural network will take in a set of data, process it, and then give an output.

1.2.6 Backpropagation

Backpropagation is a form of training a neural network to help it make the correct decision. In backpropagation, a neural network is given a set of data with the correct output known. The neural network determines its answer which is compared to the correct answer. The loss function gradient is then determined and fed back into the neural network to help it make better decisions in the future.

1.3 References

1. Ahmad, Iftikhar, Azween B. Abdullah, and Abdullah S. Alghamdi. "Application of Artificial Neural Network in Detection of DOS Attacks." *Proceedings of the 2nd International Conference on Security of Information and Networks - SIN '09* (2009): n. pag. Print.
2. Jing-Xin, Wang, Wang Zhi-Ying, and Dai Kui. "A Network Intrusion Detection System Based on the Artificial Neural Networks." *Proceedings of the 3rd International Conference on Information* Security - InfoSecu '04 (2004): n. pag. Print.
3. "KDD Cup 1999 Data." *KDD Cup 1999 Data*. N.p., n.d. Web. 24 June 2016.
4. Srivastav, N., and R. K. Challa. "Novel Intrusion Detection System Integrating Layered Framework with Neural Network." *2013 3rd IEEE International Advance Computing Conference (IACC)* (2013): n. pag. Print.

2 Project Overview

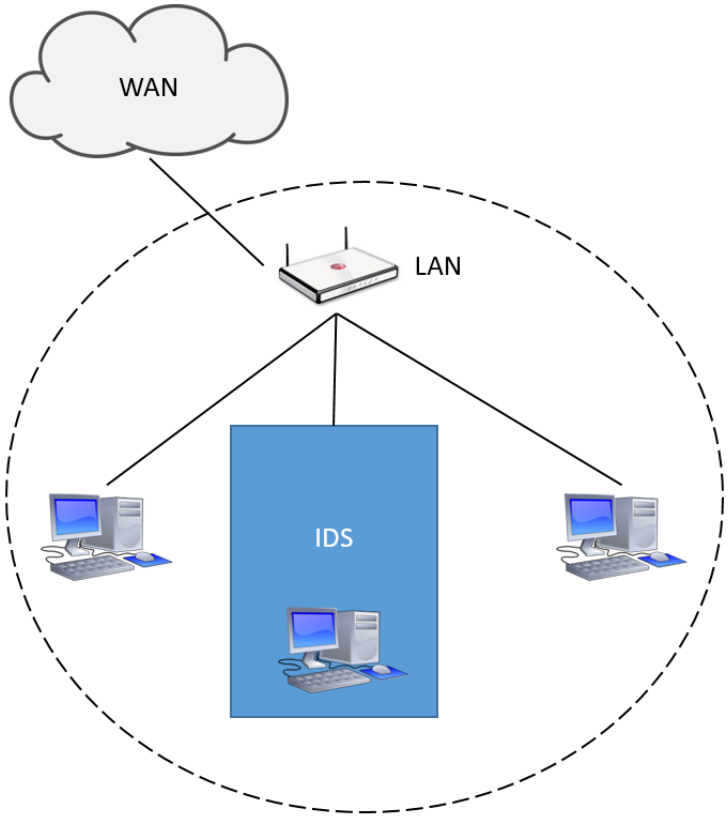

Figure 1. IDS within a Local Area Network

The Intrusion Detection System will be a host-based system. This means that the system will run on a host device and monitor traffic that the host device receives. This host device will be a computer within a LAN that is connected to a WAN. This will allow for both personal users and larger organizations to use the system easily and effectively. The figure above shows how the IDS will play a role within the LAN.

As stated, the Intrusion Detection System will monitor network traffic received by the host device. The IDS software will watch the data that is being received on the network card of the host computer and process that data by using a neural network. This neural network will consist of multiple layers. First, the data will enter through the input layer, continue to one to many hidden layers, then finally reach an output layer. The output layer will be the result which will tell the IDS if the traffic is malicious or not. Using a neural network does however mean that false positives and false negatives may arise as a result. In order to reduce the amount of these errors, the neural network will first be trained using a backpropagation technique with a set of known data from the KDD Cup 1999 Data. This data will allow the neural network to update its weights between each of the layers.

2.1 Project Goal

The goal of this project is to create a system that can detect malicious DOS attacks on a host computer. The accuracy goal for this is to reach at least 85% accuracy. Any false positive or false negatives will count against this accuracy. This accuracy will be tested on a large known dataset. The system will watch the network traffic received by the host computer, process the information and make a decision as to the intent of the traffic. If the system determines that something is malicious it shall notify the user. This system will allow for individual users and larger organizations a safer and more secure LAN.

2.2 System Context

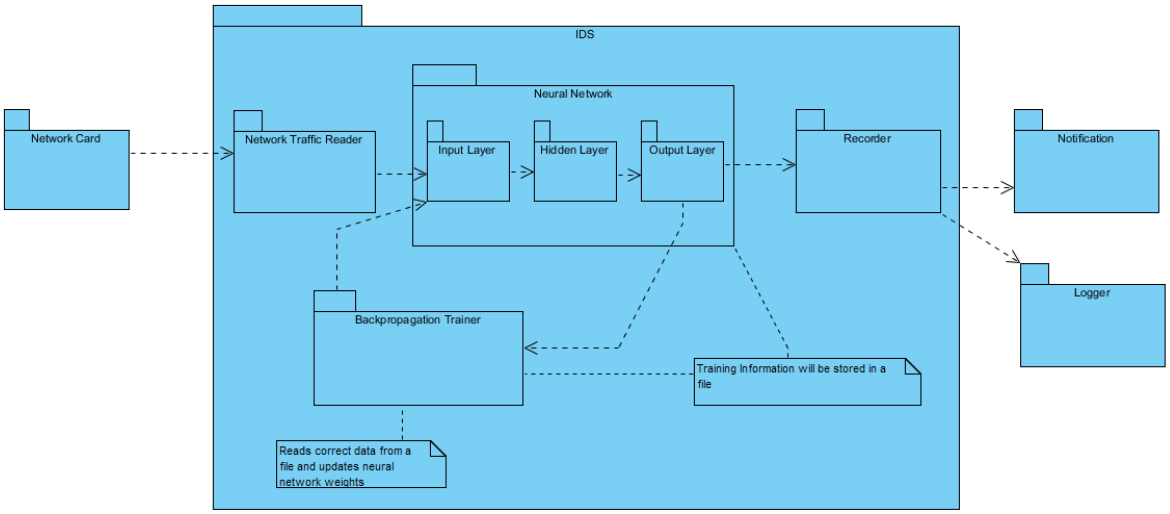

Figure 2. IDS System Context Package Diagram

The IDS will only take input from the host network card. The network traffic reader portion of the IDS will read the data and format it in a way to pass it to the neural network. The neural network will also have a file that contains its weights and other information in order to avoid training the neural network before starting the application every time. The decisions from the neural network will be passed to a recorder piece which will notify the user and log the information in a file. This allows the user to know at that moment that there may be a possible attack and to look at the information later. This allows the user to add any changes to the backpropagation trainer if desired to help avoid incorrect decisions in the future.

3 Requirements Specification

3.1 Critical Use Cases

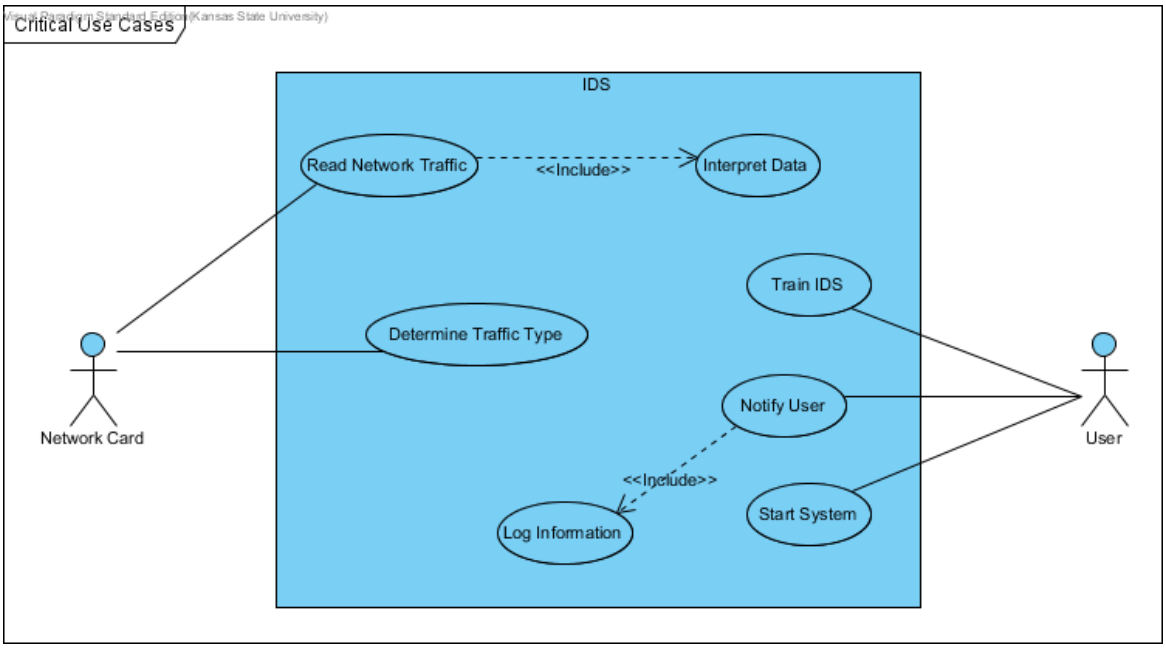

Figure 3. IDS System Critical Use Cases

3.1.1 Use Case 1: Read Network Traffic

Description: This use case describes the process of reading the network traffic data.

Includes: Interpret Data

Pre-Conditions: The IDS system must be able to interact with the network card and the computer must be connected to the internet to receive network traffic.

Details: The IDS system will read network traffic that comes in through the network card of the host computer.

Post Conditions: The IDS system will now have network traffic data in memory.

Specific Requirements:

3.1.1.1 SR1.1 [Critical Requirement]

The system shall be able to read data from the host system network card.

3.1.2 Use Case 2: Interpret Data

Description: This use case describes the process of interpreting the network traffic data into a format for the IDS system to use.

Pre-Conditions: The IDS system must have already read in some of the network traffic data into memory to be interpreted.

Details: The IDS system will read the data that it has received from the host network card and interpret the data in a way that other aspects of the system can use.

Post Conditions: The IDS system will now have network traffic data in memory that is in a format usable for other parts of the system.

Specific Requirements:

3.1.2.1 SR2.1 [Critical Requirement]

The system shall be able to interpret the data from the received network traffic and store it in a usable format.

3.1.3 Use Case 3: Determine Traffic Type

Description: This use case describes the process of determining the network traffic type.

Pre-Conditions: The IDS system must have interpreted data contained in memory. The neural network piece of the IDS must already be trained.

Details: The IDS system will read network traffic data stored in memory and determine if the traffic is malicious or not. This decision will be made by the trained neural network.

Post Conditions: The IDS system will now have a decision based on the network traffic data that it read.

Specific Requirements:

3.1.3.1 SR3.1 [Critical Requirement]

The system shall be able to determine if the network data received by the host machine is malicious with at least 85% accuracy.

3.1.3.2 SR3.2

The system shall determine what type of attack is being made to the host network when malicious network traffic is found.

3.1.4 Use Case 4: Train IDS

Description: This use case describes the process of training the IDS system from known network traffic data.

Pre-Conditions: The IDS system must be given the known network traffic data.

Details: The IDS system will read the known network traffic data and determine how accurate the decisions it makes are. The system will then

backpropagate the errors back into the neural network to train the system to make better decisions in the future.

Post Conditions: The IDS system will now be trained in order to properly make decisions from network traffic data.

Specific Requirements:

3.1.4.1 SR4.1 [Critical Requirement]

The system shall be able to train itself through backpropagation on known network traffic data.

3.1.5 Use Case 5: Notify User

Description: This use case describes the process of notifying the User of a malicious attack.

Includes: Log Information

Pre-Conditions: The IDS system must have a known malicious attack on the host system.

Details: The IDS system will notify the user that a malicious attack has been made on the host system. The notification will tell the user what type of attack has been made on the system.

Post Conditions: The User will have a visual notification of an attack and the type of attack.

Specific Requirements:

3.1.5.1 SR5.1 [Critical Requirement]

The system shall be able to notify the User of the host system when a malicious attack is encountered.

3.1.6 Use Case 6: Log Information

Description: This use case describes the process of logging malicious attacks on the host system.

Pre-Conditions: The IDS system must have a known malicious attack on the host system.

Details: The IDS system will log the malicious attack in a file for the User to read. This will allow the User to track all malicious attack on the system.

Post Conditions: The User will have access to a log file containing a list of all malicious attack on the host system.

Specific Requirements:

3.1.6.1 SR6.1

The system shall be able to log all malicious attacks into a log file.

3.1.7 Use Case 7: Start System

Description: This use case describes starting the IDS system on the host machine.

Pre-Conditions: The IDS system must have been trained to work properly, but can be started without training.

Details: The IDS system will start reading network traffic and making decisions based on the training that it has received.

Post Conditions: The IDS system will be in a running state.

3.2 Assumptions

3.2.1 The host system must be able to run a python 3.5 application.

3.2.2 The User must have a set of training data to give the IDS system to train from.

3.2.3 The host system must have a network card and be able to receive network traffic data.

3.3 Constraints

3.3.1 The IDS system may have false positives and false negatives and therefore will be constrained to only being able to make decisions based on how well it is trained.

3.3.2 The IDS system will be constrained by the host system's processing and will only be able to process quickly if the system has enough allocated resources.

3.4 Environment

3.4.1 The IDS system will be written in Python 3.5.

3.4.2 The IDS system will be developed in the Visual Studio Community IDE.

3.4.3 The IDS system will be tested using a Windows 10, 64-bit operating system.

3.4.4 The training data for the system will come from the KDD Cup 1999 Data set shown as number 4 in the references section of this document.