Intrusion detection in networks and servers

Problem Statement

An **intrusion detection system** (**IDS**) is a device or software application that monitors a network or systems for malicious activity or policy violations. Any malicious activity or violation is typically reported either to an administrator or collected centrally using a security information and event management (SIEM) system. A SIEM system combines outputs from multiple sources, and uses alarm filtering techniques to distinguish malicious activity from false alarms. So, in this we will design a new intrusion portfolio that will addresses the challenges you face every day. Intelligent detectors with false alarm immunity, keypads that are easier to use, panels with more power, and communications solutions that compensate and provide us the tools to create high-quality, reliable solutions for our customers. Support all major communication formats plus **internet & StarLink Wireless Radios**, universal primary/backup communicator (hi-speed up/downloads from Gemini Panels). Up/downloading, including unique PC-preset unattended method.

Background

Network intrusion detection systems (NIDS) are placed at a strategic point or points within the network to monitor traffic to and from all devices on the network. It performs an analysis of passing traffic on the entire subnet, and matches the traffic that is passed on the subnets to the library of known attacks. Once an attack is identified, or abnormal behavior is sensed, the alert can be sent to the administrator. An example of an NIDS would be installing it on the subnet where firewalls are located in order to see if someone is trying to break into the firewall. Ideally one would scan all inbound and outbound traffic, however doing so might create a bottleneck that would impair the overall speed of the network. OPNET and NetSim are commonly used tools for simulating network intrusion detection systems. NID Systems are also capable of comparing signatures for similar packets to link and drop harmful detected packets which have a signature matching the records in the NIDS. When we classify the design of the NIDS according to the system interactivity property, there are two types: on-line and off-line NIDS, often referred to as inline and tap mode, respectively. On-line NIDS deals with the network in real time. It analyses the Ethernet packets and applies some rules, to decide if it is an attack or not. Off-line NIDS deals with stored data and passes it through some processes to decide if it is an attack or not.

Methodology

Step 1: Data collection and dataset preparation

This will involve collection of data from various sources and formatting them, annotating them with ground truth object bounding boxes

Step 2: Developing a Anomaly-based intrusion detection systems

The basic approach is to use machine learning to create a model of trustworthy activity, and then compare new behavior against this model. Although this approach enables the detection of previously unknown attacks, it may suffer from false positives: previously unknown legitimate activity may also be classified as malicious.

Step3**:** Developing a Host intrusion detection systems

Host intrusion detection systems (HIDS) run on individual hosts or devices on the network. A HIDS monitors the inbound and outbound packets from the device only and will alert the user or administrator if suspicious activity is detected. It takes a snapshot of existing system files and matches it to the previous snapshot. If the critical system files were modified or deleted, an alert is sent to the administrator to investigate.

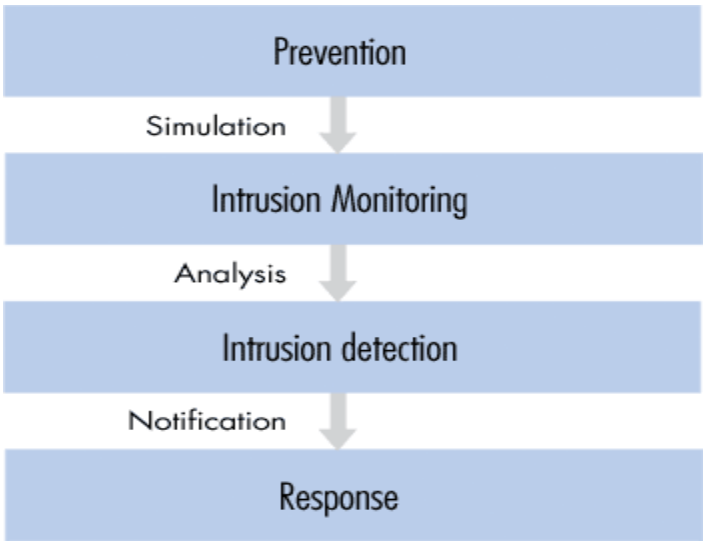

Fig 1: Architecture of Intrusion Detection system

Experimental Design

Dataset: Any data in form of numbers, alphabets etc.

Software and Hardware Requirements: Python based Computer Vision and Deep Learning libraries will be exploited for the development and experimentation of the project. Tools such as Anaconda Python, and libraries such as OpenCV, Tensorflow, and Keras will be utilized for this process. Training will be conducted on NVIDIA GPUs for training the end-to-end version of CNN based object detection model.