Intrusion Detection Systems (IDS) have been around for quite some time now, their goal is to detect and notify a network administrator of possible malicious activity on a secure network. This is usually done by routing all the network traffic through the IDS. This lets the IDS inspect the traffic and determine if any malicious activity is taking place. Traditionally, this is done with simple rules configured to the IDS. These systems can use information of previously known attack signatures or known malicious attributes to determine if something suspicious is happening within the secure system. The next step is to use Machine Learning (ML) algorithms to further enhance the abilities of these systems. When referring to ML in IDS, you can divide approaches into Artificial Intelligence (AI) methods, and Computational Intelligence (CI) methods. AI techniques refer to methods normally used by AI models, such as statistical modeling. AI are also able to process symbolic information in a way that CI cannot.[6] CI methods normally handle numeric representation of information, with methods like evolutionary computation, artificial neural networks, and artificial immune systems. For now, we will be focusing on CI methods.

One such CI method is a Genetic Algorithm (GA). GA aim to find an optimal solution to a problem. This is done by using a simplified model of biological genetics to try and mimic the process of evolution, evolving the program to best fit into its environment and serve its purpose. In the realm of intrusion detection, potential solutions to a problem are encoded as sequences of bits. These single bits are called genes, while the sequences of multiple genes together are called chromosomes. The algorithm will then test a set of chromosomes, called a population. From here the program will test this population to see how well of a solution they provide. The parameters for determining how well the solution works will vary depending on the issue at hand.

Decision trees are another method that can be used to classify data with common attributes. When implementing this into an IDS system, an administrator will have to start with providing relevant features so that the program will know what to look for when distinguishing normal behavior from potential intrusions. Features that would be most relevant are attributes such as protocol type, port number, packet length, and frequency of connections. From here, the tree will be trained on a labeled dataset where each instance is tagged as either normal operation or a specific type of attack. To avoid the tree from becoming overly complex with diminishing returns, a process such as pruning will also come into play. Pruning is where the nodes that do not contribute significantly to the overall decision-making process are removed to simplify the table. Once the decision tree has been trained and validated on known information, it can be deployed to an operational environment. As network traffic flows in, the IDS will then use the decision tree to classify the activity as good or bad in real time.[3] One of the major advantages of decision trees is their ability to be continually updated with new data to adapt to newer network traffic as well as intrusion techniques. Decision trees are favored for their simplicity as well as their interpretability. The data from this method is easily readable by a system administrator, making it much easier for an admin to double check the work of a decision tree in case of a mistake.[1] Another reason decision trees are useful is the fact that they accurately classify intrusions, with the proper data training they can be up to 96% accurate at determining if activity is malicious or if it is just regular traffic moving through the network.[2] There are many different ways of implementing ML into an IDS system, each of them having their own set of pros and cons.

[1] An application of machine learning to network intrusion detection

https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=42a7f6ebdf173dc2c642236f4485bc8445483399

[2] Machine learning techniques for intrusion detection

https://arxiv.org/pdf/1312.2177.pdf

[3] An adaptive ensemble machine learning model for intrusion detection

https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8740962

[4] Intrusion detection using machine learning

https://shorturl.at/qwxM7

[5] Machine learning algorithms in context of intrusion detection

https://www.researchgate.net/profile/Helmi-Rais/publication/311755594_Machine_learning_algorithms_in_context_of_intrusion_detection/links/5ca1e22892851cf0aea5805a/Machine-learning-algorithms-in-context-of-intrusion-detection.pdf

[6] Cyber intrusion detection using machine learning classification techniques

https://www.researchgate.net/profile/Sohrab-Hossain-2/publication/343043898_Cyber_Intrusion_Detection_Using_Machine_Learning_Classification_Techniques/links/5f1583eaa6fdcc3ed718bc7e/Cyber-Intrusion-Detection-Using-Machine-Learning-Classification-Techniques.pdf

[7] Multi-Stage Optimized machine learning framework for network intrusion detection

https://arxiv.org/pdf/2008.03297.pdf

[8] A machine learning-based intrusion detection for detecting IOT network attacks

https://www.sciencedirect.com/science/article/pii/S1110016822001570

[9] Machine learning based intrusion detection systems for IoT Applications

https://arxiv.org/pdf/2302.12452.pdf

[10] Adversarial machine learning in network intrusion detection systems

https://arxiv.org/pdf/2302.12452.pdf