Chapter: 14

GRAPH BASED METHODS TO DETECT NETWORK ATTACKS IN CYBER SECURITY

Mr. Pushpendra Kumar*
Faculty, Glocal School of Science and Technology,
Glocal University, Saharanpur, U.P.
***Correspondence to:** pushpendra@theglocaluniversity.in*

Mohd Hyder Gouri
Faculty, Glocal School of Science and Technology
Glocal University, Saharanpur, U.P.

***DOI:** https://doi.org/10.52458/9788196869434.2023.eb.grf.ch-14*
Ch.Id:-GU/GRF/EB/ETCSIA/2023/Ch-14

ABSTRACT

The advancement of 5G networks and AI technologies has introduced new cyber security challenges for wireless communication system. These challenges include potential vulnerabilities in network infrastructure, increased attack surface, and need for more sophisticated threat detection and mitigation strategies to protect sensitive data and privacy. Deep learning has certainly made strides in enhancing attack detection methods.

Keywords: *Graph, Normalization, Detection Systems, Security, Network Attack*

INTRODUCTION

In today's interconnected world, where businesses, governments, and individuals rely heavily on digital technology, cyber security is of paramount importance. Cyber security refers to the practice of protecting computer systems, networks, and digital information from theft, damage, or unauthorized access. It encompasses a wide range of technologies, processes, and practices designed to safeguard against cyber threats. Focuses on protecting the integrity, confidentiality, and availability of data as it flows across a network. It involves safeguarding data, whether it's stored, transmitted, or processed, from unauthorized access, disclosure, alteration, or destruction to ensure privacy and integrity. Encryption, access controls, and data masking are common techniques. An internetwork, also known as an internet, is a collection of interconnected networks that use a common set of communication protocols. The most well-known example of an internetwork is the global public internet. A graphical representation of an internetwork typically involves nodes (representing networks or devices) and edges (representing connections between them). Here's one of the most important challenges is that to monitor suspicious activity in graphical network which are connected through communication link(path). with the help of path data packet moves from source to destination. For detect suspect data packet those data packet may loss our important data and fetch prominent information from our system or system may be damage. Therefore, to detect these data packet we use some efficient Graph based method like as attack graphs and attack trees are the most popular method.

LITERATURE REVIEW

Attack Graph: A cyber-attack graph is a representation of all possible paths of attack against a cyber security network, illustrating a state where an attacker has completed a successful breach. There are two popular forms of attack graphs. The first is a direct graph where nodes represent network states and edges represent exploits that transform one state into a more compromised state, ultimately showing a successful attack. A second form is a direct graph where nodes represent pre- or post-conditions of an exploit, and edges represent the consequences of having a pre-condition that enables an exploit post-condition.

The main purpose of this project has been to study the potential of GNNs in network attack detection. GNNs are a novel family of traditional Neural NNs, which start from a different conceptual basis, they operate over graph-structured data. Although GNNs have been researched in topics such as chemistry or physics, their application in the domain of cyber security is still under observation. That is why, one of the main contributions of this project has been to shed a light on the application of GNNs to network attack detection, to increase trust in their application and incentivize further research.

Intrusion Detection Systems (IDs): There are different types of IDS systems, which vary according to the setup where they are deployed. Those deployed in a particular host are known as host-based, whereas those deployed in a computer network are known as network-based. Both types have advantages and disadvantages, however, for the scope of this project, although we will introduce host-based systems, we will focus on the network-based type.

Detection Methods: An IDS can leverage different detection methods to find threats or policy violations. There are mainly two types: signature-based, which works on the basis of looking for patterns of already known attacks, and anomaly-based, which relies on modelling the normal behaviour of the network and looking for events which differ from this normality.

Signature-Based: As the name suggests, this detection technique is based on signatures 4 of known attacks, which means that the patterns found in the data are cross-checked against records of already-seen attacks. This method is similar to antivirus software, which leverages a database or record of past attacks to match any potential existing threats.

For instance, if certain IP addresses are blacklisted, a signature-based IDS will detect any traffic from these addresses and raise the corresponding alerts. Or if an attack matches a well-known pattern, such as a flooding attack, the IDS will recognize this known pattern and act accordingly.

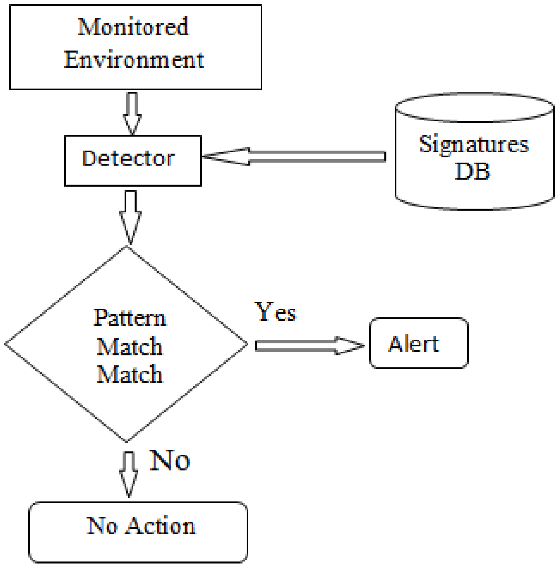

Figure-1: Signature-based detection method flowchart

Signature-based IDS have the main advantage that they are robust against known malware, however, it is clear to see its main limitation, they are vulnerable to zero-day attacks or any type of malware not recorded or known.

Anomaly-based: The fundamental basis of this detection method is to identify anomalous behaviours in the inspected traffic, which was introduced as a new detection method due to the rapid evolution of malware attacks. These systems mainly use ML techniques to create a trustful behaviour model of the network, and when something differs from the established model, it can be treated as an anomaly or a potential threat.

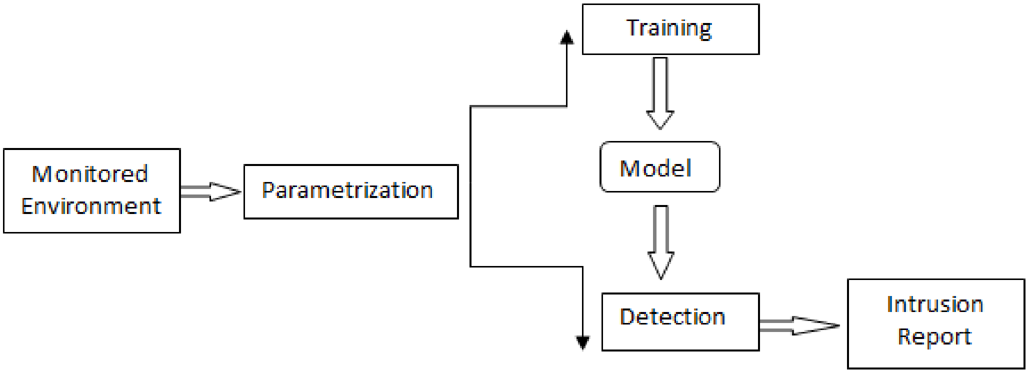

Figure-2: Anomaly-based detection method flowchart

As can be seen in Figure-2 shown above, they work like most ML models. Initially, the monitored environment is parametrized and data is collected to create training datasets. Once the ML model is built, it can be used on validation data to test its performance against unseen scenarios. This process has to be iteratively repeated, as it is not enough to train a model once, we have to keep it updated by periodically feeding it with new training datasets. So, even if we achieve high performance with certain data, our model may not be as accurate when tested with totally different data, or may also become obsolete very fast.

Hence, anomaly-based models may prove to perform well in a test environment, but face problems when exposed to production data.

This could mainly happen due to three different reasons:

- **Lack of generalization:** These models lose prediction capabilities when exposed to new network scenarios and traffic.
- **Overfitting:** Most proposals present very high classification results, but this is normally due to overfitting of the training datasets, which makes the model vulnerable to variations of attacks or new types of malwares.
- **Features:** These models may be based on a certain set of features which are network dependent and cannot be extrapolated to other setups and conditions. The main challenge faced is to develop an anomaly-based system capable of performing well in any type of environment. To achieve this, we need to find a solution which tackles the three aforementioned limitations.

This means that we need to carefully select a set of features which can be applied in different network scenarios and build a model that does not create overfitting with the training data. That is why we proposed the use of GNNs, as these are based on certain theoretical concepts which could prove to solve these limitations.

CONCLUSION

Networks and security are one of them, as network data and traffic can be naturally represented as a graph. Existing approaches analyze network flows individually, effectively obviating the fact that flows present inter-dependencies between them. That is why this novel approach can unlock and create tremendous value. The different network events, especially the network attacks, should be graphically distinguishable, as differentiability between graphs will favor the GNN model in the classification process. That is why, the dataset used should be previously deeply inspected to create a graph representation tailored to the contents of the data. In our case, we came up with a three-node representation that helped distinguish between PortScan, DDoS and benign traffic. Through the different hypothesis-validation experiments, we came up with the conclusion that solely the graph structure is not enough to classify the network events, as we also need some relevant features information.

REFERENCES

1. *Hung-Jen Liao, Chun-Hung Richard Lin, Ying-Chih Lin, and Kuang-YuanTong. Intrusion detection system: A comprehensive review. Journal of Network and Computer Applications*

2. *Benjamin Sanchez-Lengeling, Emily Reif, Adam Pearce, and Alexander B. Wiltschko. An introduction to Graph Neural Networks.*

3. *M. Gori, G. Monfardini, and F. Scarselli. A new model for learning in graph domains. IEEE International Joint Conference on Neural Networks,*

4. *Miquel Ferriol-Galmés, José Suárez-Varela, Jordi Paillissé, Xiang Shi, Shihan Xiao, Xiangle Cheng, Pere Barlet-Ros, and Albert Cabellos-Aparicio. Building a Digital Twin for network optimization using Graph Neural Networks. Computer Networks: The International Journal of Computer and Telecommunications Networking, Volume 217*

5. *Krzysztof Rusek, José Suárez-Varela, Albert Mestres, Pere Barlet-Ros, and Albert Cabellos-Aparicio. Unveiling the potential of Graph Neural Networks for network modelling and optimization in SDN. Proceedings of the ACM Symposium on SDN Research (SOSR).*