Cyber Attacks Visualization and Prediction in Complex Multi-Stage Network

Mr. Jassir Adel Altheyabi

Master's in Cyber Security, Cyber Security Department, Majmaah University,
Kingdom of Saudi Arabia

Email: Eng.Jasir@Gmail.com

Abstract:

In network security, various protocols exist, but these cannot be said to be secure. Moreover, is not easy to train the end-users, and this process is time-consuming as well. It can be said this way, that it takes much time for an individual to become a good cybersecurity professional. Many hackers and illegal agents try to take advantage of the vulnerabilities through various incremental penetrations that can compromise the critical systems. The conventional tools available for this purpose are not enough to handle things as desired. Risks are always present, and with dynamically evolving networks, they are very likely to lead to serious incidents. This research work has proposed a model to visualize and predict cyber-attacks in complex, multilayered networks. The calculation will correspond to the cyber software vulnerabilities in the networks within the specific domain. All the available network security conditions and the possible places where an attacker can exploit the system are summarized. The vulnerability-based multi-graph technique for the attacker is presented using a matrix. Also, an attack graph algorithm is proposed, leading to the identification of all the vulnerable paths that can be used to harden the network by placing sensors at the desired locations. The presented attack graph is used for vulnerability assessment of Multi-Stage Cyber Attacks.

Keywords: Network vulnerability; attack graph; adjacency matrix; clustering technique; cyber defense.

1. Introduction

In recent years, there is effective use of the Internet and related technologies. Identifying different types of models that are responsible for providing services based on the internet and network can be said to be increasing day by day. With the use of the internet, the amount of data collected on servers and network computers is increasing significantly. The availability of vulnerable and critical data on the systems makes it very easy for the attacker, and the breach of data impacts companies' intellectual property (Mishra et al., 2021). The threat of information leakage or attack on network-based devices is increasing day by day. It is high time to figure out how these attacks can be mitigated or rather prevented and thus programmers all over the world are trying to develop and design systems that are capable of detecting the intrusion of non-ethical attackers from remote locations (Gupta et al., 2020). A large number of cybercrimes are being reported every now and then and based on these, vulnerabilities are identified (Mishra et al., 2020), (Mishra et al., 2021). Intrusion detection can be done in two ways: an intrusion detection system and an intrusion prevention system. However, the prevention system only assigns the risk of getting an exploit while on the other hand, the detection system can raise the alarm and detect a threat to the software-based system or server machines. The availability of such systems is indeed the need of the hour, and more secure mechanisms are required to be designed as well as developed to protect the privacy and integrity of data of individuals (Sarker et al., 2020), (Mishra, 2020). As a general approach, the intrusion detection system is more efficient and categorized based on specific parameters that are:

Analyzed Activities: The system that can analyze the activities taking place in the network and at the same time also detects the involvement is taken under the purview of activity-based intrusion detection system.

Host Intrusion Detection System: The system that can get attached to the host machine and thereafter formulate an identifier is taken under the purview of host intrusion detection systems.

Both the above methods use either signature-based activity or host detection, or anomaly-based detection. Depending on host's signature trying to communicate on the network, intrusion can be detected, or anomaly-based detection can be used to detect intrusion.

However, in both cases, it is impossible to identify a specific type of intrusion or predict how the network path may manage itself at the time of intrusion (Aldweesh et al., 2020).

In this paper intrusion, detection systems that are effective to identify attacks or exploits in the network have been discussed upon.

The main objective of this research is to identify an approach that can calculate all attack locations in any computer network.

There are various ways in which the attack location can be identified by making use of a graph, and other such similar approaches can be taken. The optimal placement of the graph in the network is helpful enough to identify and provide the positions where the attack is most likely to happen.

In this research, the proposed model performs both visualization as well as prediction of the attacks. The computation corresponds to the cyber software vulnerabilities in the networks within the specific domain. The attacker will try to exploit the ocean of the network that is vulnerable to attacks. The central ideology behind this particular approach is to identify the gray areas responsible for any exploitation of the network by the attacker. For this purpose, traditional graph-centric modeling and adjacency matrix have been used. Also, given all known attack areas of the network, an attacker will always try to identify and evaluate all locations in the network path that provide opportunities for exploitation. This approach continues with an enhanced version of network hardening, that identifies areas where exploitation is possible and implements prediction parameters that include specific sensor-based investigations. This makes correlation and prediction very simple that in turn helps secure the network in a better way. The attack graphs can be created and examined for further placement of vulnerability assessment tools. The visualization of the attack graphs can be done using various visualization techniques that are available these days.

The paper is structured in the following way:

The literature of attack graph, a type of vulnerability-based multi-graph technique for the attacker, is discussed in section 2. Proposed model, vulnerability assessment, adjacency matrix, clustering algorithm in homogeneous groups, multi-Step attack path identification, and detected intruders are presented in section 3. Section 4 discusses the results and section 5 concludes the paper with some of the future directions.

2. Literature Review

In recent years, there have been several advancements in attack graphs as a result of which it is now possible to create an attack graph or realistic computer networks. For a long time, researchers have been trying to process the information relevant to attack graphs and their creation, along with real-time network systems (Husak et al., 2018). We have studied many such techniques for the composition of our approach to make it very effective (Liu et al., 2020).

The methods presented in this research mainly focused on creating the adjacency matrix from the attack graphs (Ghadi et al., 2020). A similar author also gave a robust report on vulnerability assessment using attack graphs. In (Liu et al., 2020), the authors presented a very strong approach to network security and proactive prevention of intrusion at various stages. According to the details prescribed in (Lallie et al., 2020), the vulnerability was one of the most important reasons for cyber-attack. The identification of such vulnerabilities was most important and presented in the study indicated by (Pirani et al., 2021). Another critical part of our project is the visualization technique for the attack graphs. The prescribed theory proposed by (Cinque et al., 2020) formed the basis for graph visualization. In (Stergiopoulos et al., 2021), the authors presented another fantastic technique for visualization, which was integrated into one of the research methodology modules in this study. The most crucial part of our research is the placement of appropriate sensor-based devices at vulnerable positions. Identification of such vulnerable positions is another important part required for the study. A schematic representation is possible for placing sensors at a suitable position for alerting and prioritizing in the attack graphs.

The author proposed a robust design for placing sensors in the attack graphs for appropriate responses in cyberspace (Pourhabibi et al., 2020). The next phase of the research deals with identifying attack points and hardening the network to avoid vulnerability. The attack graph hardening techniques prescribed in the study (Ibrahim et al., 2020), (Sansavini & Parigi, 2020) that makes use of topological analysis gives a new dimension for managing cyber threats at the time of vulnerabilities. Thus, topological analysis for vulnerabilities became the backbone of the study and proposal. A structured method for finding the correlations between the intrusion events and the attack scenario is with the help of attack graphs. However, these graphs are considered to be very complex, and the hierarchical aggregation of these graphs was presented in (Singhal & Ou, 2017).

The methodology proposed in the research (Stergiopoulos et al., 2021), (Yang et al., 2019), (Ramadan, 2020), has proven to be a positive solution to various network-based vulnerability and exploitation problems. The ideology derived from this research, focused on identifying vulnerabilities and providing a smooth mechanism to eliminate these problems at various stages.

The proper results for the proposed research can only be achieved with the help of the entire architecture. As described in the studies of (Pirani et al., 2021), the comprehensibility of the complex graph for the networks can be done with the help of the algorithm proposed by the researcher.

Each edge of the graph represents a network path in the proposed study and the interconnectivity of the different edges in the network graph is represented in the form of an adjacency matrix, which becomes the second important pillar of the study of (Lallie et al., 2020), (Pirani et al., 2021), (Russo et al., 2019). However, the analysis of the complex graph alone is not enough to identify the gray areas. Thus, to find a suitable solution for the same, it was vital for us to find an appropriate means to visualize the data (Ivanov et al., 2020).

In (Medvedev et al., 2021), the authors proposed a very effective ideology for visualizing the data from the graph in a systematic manner which will be helpful enough to identify the network vulnerabilities.

With the help of this particular information, it will be very easy for the algorithm as well as the methodology to identify the problems and predict the type of network problems that might arise (Nia et al., 2019). The approach that has been proposed in the method tries to give a probable solution to the problem that may occur due to vulnerabilities in a particular network when there are opportunities for exploits. In the last two decades, the issue of cybersecurity has become most significant. A large number of researchers have come up with various protocols and techniques to identify and deal with these problems (Das et al., 2021).

Many studies have been considered, and thereafter a new methodology for network security applications based on complex attack graphs has been derived (Lallie et al., 2020). A large number of operations were studied to find out the regularities in the graphs, which could be a probable cause for graph clustering that can lead to changes in network configuration. Various research done by scholars from different parts of the world have been studied to visualize data from the graphs (Ghazo et al., 2019). The appropriate visualization of the attack graphs and their adjacency matrix formed the foundation of this research, as well as the provision of an appropriate suitable format in which this can be studied very easily.

Identifying all exploitation parts from the graph using various prediction techniques coupled with the underlying principle of vulnerability and evaluation makes this approach a game-changer in the realms of network security.

The approach used in this particular research develops a technique for identifying complex attack graphs. It is undeniable that in large networks a large number of densely connected subgraphs occur. However, most of the vulnerabilities in the network tend to occur in this particular subgraph. The adjacency matrix for the corresponding attack graphs represents the edges that are most vulnerable to network attacks.

An information-theoretic clustering technique is applied to identify the branches of the adjacency matrix in the graph. However, the clustering technique is assumed to be fully automatic, which requires linear scaling. In general, the elements of the adjacency matrix represent the steps during an attack. With the help of these edges, the reachability of the attacker can be identified. After identifying the various factors, a single matrix can be derived. The method presented in the research tries to deal with all the possible steps and the part that is a transitive closure of the attacker's graph. The method used in this particular research considers the vulnerability-based graph attack. All the available network security conditions and the possible places where an attacker can exploit the system are summarized, with the help of this representation.

With this the attacker's attack patterns can be made clear. The attacker represents the vulnerability-based multi-graph technique with the help of a matrix. However, the approach is straightforward, yet there are chances for anomaly or complexity to arise as a general scheme.

As a general scheme. It can be seen that the following assumptions are possible when applying the vulnerability assessment approach in attack graphs.

- The network is assumed to be finite.
- The vulnerability assessment is performed using the attack graph.
- It is assumed that the analysis of the graph NP is complete.
- The approach can be used to identify an optimal solution for network hardening.
- It is not mandatory that all the paths of the network are identified.
- The prediction is not intended to be complete and there may be other solutions for a particular network path.
- Penetration testing or exploitation is not an integral part of this methodology.

- The analysis and synthesis of the attack graph are done based on the parameters provided by the network.
- The vulnerability assessment is intended to provide the grey areas where a solution is required. However, it can be further refined and simplified to provide more accurate results. The accuracy of the approach varies from network to network and can be further simplified by improving graph analysis algorithms.

3. Research Methods

The research is divided into the main elements for creating the attack graphs, identifying the matrix, placing intrusion detection devices in appropriate locations, and finally making the network security metrics for the cyber-attacks.

1. This leads to the creation of the attack graphs using the vulnerability assessment in the topologies.

2. This leads to our noble approach responsible for applying adjacency matrix criteria for cyber-attack graph analysis.

3. Gives the placement positions for various network intrusion detection devices in the particular network for which the cyber-attack graphs are generated.

4.Provides us the final and effective method as per the research for identifying the exploitation, locations, and placement of network intrusion detection devices using the attack graphs.

Vulnerability assessment can be performed by creating cyber-attack graphs. An associative step in this particular milestone could be identifying network security using the attack graphs if possible, in the preliminary stage. The second milestone is the identification of the attack graph matrix. It can be created using different types of cyber-attack graph techniques listed in the background section. The matrix clustering algorithm will be helpful enough to find out different edges to identify the adjacency matrix. Finally, the transformation of the adjacency matrix can be done to represent different multi-stage attacks that are possible for the exploit domains. Prediction is based on attack and impact using a reachability matrix from attack graphs. According to the adjacency matrix and attack graphs, the placement of attack-based sensors responsible for identifying and tracking any type of attack in the network is the third important step in this particular research.

The identification of optimal sensor placement can be done using various optimization algorithms.

In this approach, a new and effective method for identifying placement positions using attack graphs has been proposed. The last important milestone for this research will be the identification of security metrics derived using attack graphs. This matrix will identify and provide the full security value based on the attack graphs created in the first and second steps, by using the above approach, the attack graph module will be built.

3.1 Vulnerability Assessment

The main backbone of this ideology revolves around vulnerability assessment. The assessment we make regarding vulnerabilities depends on the topography in which all the network elements are arranged. The analysis of all the vulnerabilities and interdependencies is done with the help of a graph. Some researchers have used this kind of approach to identify the vulnerabilities based on the topology.

The important part states that all the connected elements to the network are checked for vulnerabilities in network configuration, software configuration, connectivity, parameters, and hardware configurations. Cross-mapping is performed using network-based vulnerability sections where an intrusion is likely to occur. The custom scenarios are targeted and checked for vulnerabilities. The attack graph created using all, the topological assessments of the vulnerability mainly, focuses on the ways of penetration in a network. This provides a very proactive approach where all the sections that are likely to be prone to vulnerabilities are considered. It makes intrusion detection more effective at these points, and it is possible to respond very well to the attacks.

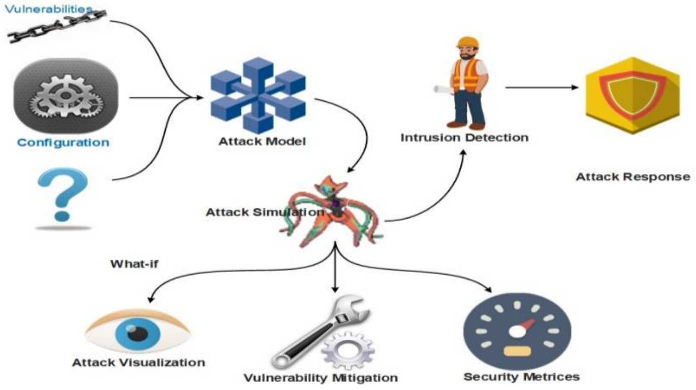

Figure 1: Overview of the Situation

A generic overview and configuration of the network and the exploitation by the hacker is shown in Fig. 1.

Hacker wants to take advantage of the fact that the vulnerabilities can be exploited once the configurations of the network are available and penetration is possible. The attack graph created from the situation is used for decision-making, which will help to find out the optimal priorities for all the hardening factors responsible for the vulnerabilities in the network. The attack graph is used to model the vulnerabilities of the systems and their potential exploits. Various strategies can be identified and optimized using the attack graph, based on the scenarios shown in the figure above. The remaining part of the graph will provide all the information needed to identify an intrusion at a particular point and all the flexible issues vulnerable to that intrusion. The attack model used by the hacker is mapped with the simulation available in the model and provides scenarios such as attack visualization, vulnerability mitigation, or security metrics based on the model used. In turn, intrusion detection can be performed, and an appropriate attack response mechanism can be designed to respond to the intruder's attack model.

The attack graph that is created can also help identify a multi-stage attack parameter for the network intrusion. Fig. 2. shows a sample network for generating the attack graph.

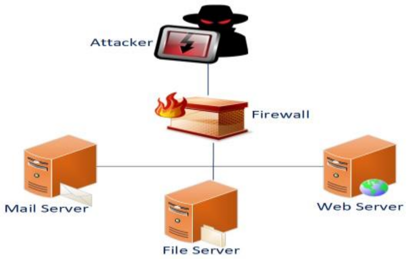

Figure 2: Sample Network for the Generating the Attack Graph

A small network is assumed to be available as a demonstration for the attack graph generation system. It is assumed that the main server and the file server are used for internal purposes, and the webserver allows connections coming from outside the network. The firewall blocks all traffic coming from outside the network and allows only those that are potentially secure connections. Considering the scenario, it can be seen whether an attacker could come from outside and compromise the mail server. To simulate this particular scenario with the creation of the attack graph, there is a need to identify the configuration elements on the network for the intrusion. There may be vulnerable software on the systems on the host computer that can act as a vulnerable device.

The most important part is that it depends on the security tool that is available on the network and used for scanning vulnerabilities or attacks.

This can be considered as the most important meaning and limitation of the model that is proposed. With the help of this technique, the hacker can access the victim's machine in the network irrespective of any particular method or exploit mechanism.

Scanning of the system is done with the help of a firewall to capture present vulnerabilities, when an attacker tries to enter the internal network. As an alternative technique, the firewall rules can be processed to build the network model to ensure a high level of security. The first and most important step that the attacker will perform in a particular network is to find out the devices that are vulnerable and have some area in the configuration that can be exploited. Once the attacker has penetrated the network, he can execute malicious code and exploit the network's vulnerabilities. Even the firewall can sometimes identify such critical points and then place the compromise text for the information. It can also be noted that even if a single vulnerability is identified in the network, it can save and protect other attacks in the present scenario. Generally, when there are various methods and packages possibly for penetrating the network, this model is limited to identifying a particular critical path that can be a threat source to the entire network,

excluding all other machines. In the above figure, the file server is not an integral part of the exploit, hence it is removed. As it is not an integral part of vulnerability analysis using topology. The attack graph for a compromised mail server is shown in Fig. 3. There is no direct path from the attacker to the file server in our network scenario that would connect any kind of vulnerability in the graph.

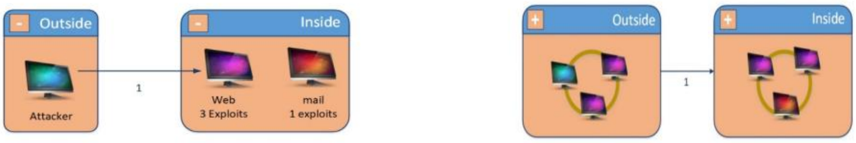

Figure 3: Attack Graph for a Compromised Mail Server in the demonstrative Network

The above figure represents the first stage of the attack graph, which can compromise the state of the mail server. Exposing the vulnerabilities possible in the firewall leads to a high level of exploitation by the firewall outside the network. In general, this exposure graph can be considered to provide information that a vulnerability originates from outside the network and attempts to access and exploit the system. Using this model, the attacker can execute any arbitrary type of suitable software in the network.

The above example shows a simple host available in the network which can be exploited using various techniques. The web server and the main server which is shown in the picture above can be a victim of exploitation multiple times. In a direct method, it is not possible to exploit a particular node. The compromise takes place. Perhaps with a sequence of steps followed by the attacker, or it can be done using multiple steps of exploits to determine the complete information by the attacker.

The vulnerability in the evaluation done using topology states many possible methods and paths to reach a particular node. Vulnerability assessment using topology is an approach where simulation is performed based on the network model. It depends heavily on identifying and collecting all the information for network configuration. The security of the network does not depend on single-time analysis. It is an interrelated process that takes place over a long period. The most important parts are protection, detection, final response, and providing the necessary inputs to the vulnerability. The most important is the discovery phase, where we try to identify which particular devices in instances are vulnerable.

However, a large number of devices cannot be detected, which is on the part of critical network-based configurations. After identifying the compromised path, further planning can be done. However, to reduce the attacks, scan with prior knowledge of the vulnerabilities.

There are many statistical analysis techniques, but the concept of network hardening is said to be one of the most optimal techniques that can be used in conjunction with attack graph analysis. Network hardening can be done using the information available from a multi-level graph. All the threats that can be identified for each type of network intrusion can be mapped. For larger networks however, hardening the network at the first step of attack exploitation reduces network exploitation in the initial stage. Initial level hardening and the consequences are shown in Fig. 4.

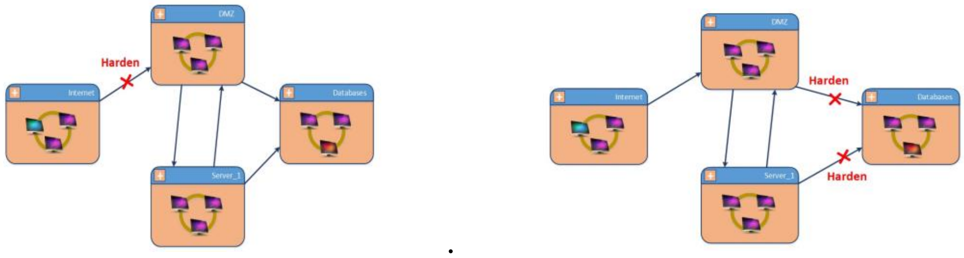

Figure 4: Initial Level hardening and the consequences

The consequences of not hardening a particular first layer analysis may lead to internal exploits by the attacker. However, it can be further reduced when hardening at subsequent layers in the network.

Thus, hardening at the final layer may also be independent of the source of the attack. In general, the particular potential attacker can be exploited at any layer without prior notice. A good network vulnerability assessment technique will always try to protect this type of exploitation at any layer. Fig.4 depicts, the initial layer of connection to the network can be hardened from the outside world so that the chances of exploitation can be minimized using this particular technique. As in a general identification scheme for first-level defense, it becomes straightforward for identification and proactive defense to avoid any vulnerability occurring in the network at any point. However, the idea is not to exploit all inputs from outside the network, but only to prevent the attacker from finding out any possible level inside the network.

Security metrics consist of the vertices and edges of the graphs available from the network vulnerability assessment.

The size of the attack graph is only a basic parameter and an indicator quantifying the effort required against the attacks. To be more precise, attack graphs are not meant to provide security against and exploitation. Instead, they are recommended to secure the network against further exploitation and block the current problems associated with first layer security.

The normalization of the matrix was created using the information for the edges and patches within the attack graph can gave a good measure to reduce the uncertainties within the graph. So, with the help of creating the security matrix, we can protect the network from any kind of compromise situation. The remedial actions can be identified and ranked in terms of risk to maximize the security and minimize the cost.

3.2 General Scenario of the Model

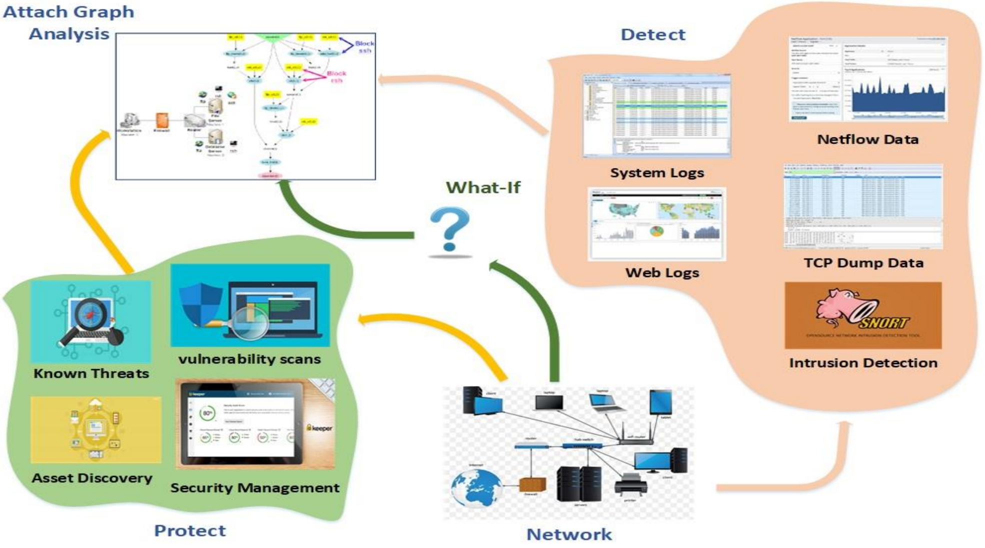

Figure 5: General Scenario of the Model

Attack graphs can identify and provide the path of vulnerabilities and protect the optimal solutions required for a particular critical network vulnerable to attack. With the help of these attack graphs, optimal decisions are made to protect the network. However, for identifying the vulnerabilities and patching the required path in the attack graph, the remaining patches in the attack graph need to be identified. The information provided by this topology-based vulnerability assessment gives us an appropriate approach for the required proactivity to rescue the forces of the attacks that are possible on a compromised path. The general scenario of the model is shown in Fig. 5.

3.2.1 Creation of Adjacency Matrix

These attack graphs can be created using specified start and endpoints, depending on the Intrusion Alarm: relationship. The graph's vertices can consist of either the attacker exploit or the network security parameters and can be aggregated into a single matrix.

The proposed solution can handle all situations where either the attacker exploits or intrusion alarms are responsible for handling the situation. If we simply consider a network consisting of about 100 machines, the total number of applied vertices for a fully connected graph according to graph theory is 100 X 100. However, while drawing this attack graph, it will not be feasible to manage and handle all possible edges. So, in general, the total length of the matrix is 100 square matrixes.

In general, for a network with n machines, the adjacency matrix will have n X n values. For simplicity, if we assume that A is a matrix consisting of all edges from our text node i to j, then the element aij represents the matrix element connecting the two nodes. To minimize the resulting matrix, we can either represent the total edges present in the matrix with a value of 1 and all values with no connection or edge with 0. Another type of data structure called an adjacency list is responsible for this and is very useful. The only simple reason for this is that the list contains all edges and the vertices that have a connection, rather than managing a complete matrix with n X n elements.

3.2.2 Clustering Algorithm in Homogenous Groups

The machines represented by the vertices correspond to the same subnet or desired scheme that is in need. The nomenclature depends on the individual perspective for constructing the attack graphs and matrix. In general, it does not depend on the order of rows and columns, but it makes sense when we talk about clusters as a whole. Therefore, applying a clustering algorithm for this square matrix is required, similar to the one proposed by (C.Ma et al., 2021). The clustering algorithm can identify the presence of high and loose density clusters and the matrix. The locations where the clusters are highly dense provide information-theoretic optimality. According to the minimum description length principle of (Hu et al., 2020), data compression can be performed for the considered clusters. Alternatively, one can also represent this fact, since the compression of the data is well understood by the sense of the regularities captured.

In general, the idea revolves around the phenomenon of clustering in the matrix for a large number of elements that are part of the network. Once the minimum description length is identified, we can protect the cluster density and the intervention probability.

3.2.3 Multi-Step Attack Path Identification

In the previous section, we illustrated how the matrix creation could be done based on the different nodes present in the network. The attack graph is mapped to a matrix that is square in shape and order. This adjacency matrix that can be created represents all the edges of the connected network attack graphs. If we assume that matrix A is constructed with a square order, it will include all possible edges from one network port to another. For a square matrix with n elements, there are n x n possibilities that can be raised as Ap, Where we can assume that its value will be A A A ....A (P times). This can be represented mathematically in the form as equation number (1).

$A_p = A. A. A. A\ldots\ldots A_p$ (1)

Let us now considered the matrix is raised to the power two, that can be represented as:

$(A^2)_{i_J} = \sum_k a_{ik} \cdot a_{k_j}$ (2)

As represented in the second equation, all the matching rows and the columns in the matrix multiplication will correspond to a particular matching step in the attack graph.

This type of submission can be done with the help of matching steps, and it leads to the discovery of the fact that each element of the matrix, obtained by the multiplication of the matrix A, will help us to identify the elements, responsible for a two-step attack between various bears as well as corresponding row and column of the attack graph matrix. The identification of A3 is going to lead us to the three-step attack elements. Thus, all the elements after the multiplication will comprise the intersection of a (i X j) value. As a rule, it can be reduced that for the multiplication of the adjacency matrix with the power n, the elements for the n- step attack graph can be identified. T multiplication for the same will provide arbitrary power that involves spectral decomposition for the matrix. Any Square matrix of the order n X n is supposed to have an eigenvalue that satisfies the eigenvalue equation: AV = VD

$A=VDV^{\wedge(-1)}$ (3)

In this equation, D represents the diagonal matrix. The eigenvectors V, corresponding to matrix A, can be calculated with the values of the elements available in the matrix. It is really simple and effective to prove that $Ap = VDpV^{-1}$

As a result, identifying the diagonal matrix for the pth power of matrix A will be very simple. The final representation for the diagonal matrix can be given as below:

$$D^p = \begin{bmatrix} d_1^p & 0 & \ldots & 0 \\ 0 & d_2^p & \cdots & . \\ \vdots & & \ddots & 0 \\ 0 & \ldots & \cdots & \end{bmatrix} \qquad (4)$$

The noble idea in this research represents that the multiplication of the matrix will be done to identify the Boolean product, irrespective of the numeric product value. According to a simple theory, the final matrix identified after the product is taken will provide at least a peace step attack from one node in the matrix to another. The identified path based on the Boolean sum of the values identified can be given by:

$$A \vee A^2 \vee A^3 \vee \dots . A^{n-1} \qquad (5)$$

With the help of the classical Floyd Warshall algorithm, we can easily identify the close value of matrix A. However, more, better algorithms are proposed by (Liu et al., 2020), (Chen et al., 2020) for a similar task. As a simple approach. We can use that classical algorithm to find out the transitive closure of the matrix. It can be further well-identified that the number of elements will increase monotonically as the increasing value of p. This means that with the large number of steps we are trying to predict; larger elements will be included in the process. So, the minimum number of steps that can be required to reach particular attack graph vertices can be easily computed with the help of identifying the reachability matrix such as:

$$A + A^2 + A^3 + \cdots + A^{n-1} \qquad (6)$$

It should be well noted that the addition, here is the Boolean addition, and the multiplication is Boolean multiplication. All the steps that are considered in this research, make use of the Boolean arithmetic laws, irrespective of the classical mathematical rules. So, as a final golden rule, equation number 6, will give us the minimum possible steps that one attacker must attack.

3.2.4 Identification of the Detected Intrusions

In the previous two sections, we described the techniques used to create a possible version of the attack graph and determined the mathematical justification for the adjacency matrix. At this stage, we can identify the locations which are more vulnerable to attacks.

In this section, we will try to summarize all the possibilities and identify the optimal placement of sensors to detect intrusion using the attack graphs.

Network intrusion is possible at any point, but the critical assets must be secured in one of the two ways. To reduce the complexity of the analysis of identifying the situations and positions that are vulnerable to the work of the network model, we try to find out the points that are more vulnerable to the attacks.

And the placement of an appropriate sensor-based alarm system is done when an attack is identified at these points. However, it remains a myth that not all parts are burglar-proof. But the most critical parts that are present through the analysis of our research can be secured. For large computer networks. Vulnerabilities are very common. Based on the design and expansion Asiatic of the software that runs on the machines. After plugging the machine into the network, it becomes very risky and vulnerable to vulnerabilities. In terms, the traditional way of maintaining the network has been done with the help of placing sensors, but it is not at all the optimal solution to minimize the hardware cost of sensors and maintenance work as well. It can also be well realized that the ever-growing universe of greed and computing resources makes it very difficult to size the network.

With the advent of cloud computing technology, it has become even more challenging to identify the data and hardware boundaries, irrespective of global sharing. Thus, it becomes problematic and impossible to manage our wide area network with the help of sensor placements at all nodes. Indeed, the software responsible for device security is not complete. Even the most sophisticated software sometimes fails to detect vulnerabilities and attacks from a remote location. So, it becomes challenging to protect the entire network and all the machines located in that particular area. Many organizations are crazy about identifying malicious activities and attacks on critical data and data centers. Sometimes, network traffic and intrusion detection at higher rates with false alarm systems make it difficult for the organization to handle the hosted information.

However, with the help of vulnerability assessment, the impact of the attacks can be reduced to a larger extent. Previously, large number of traditional tools were used to identify the computer in the network and report the vulnerabilities of the computer. However, this becomes ~~a~~ labor-intensive work and is very error-prone. While all the computers are connected and every computer is identified for vulnerability combinations. The key idea for this research revolves around the fact that they prefer to focus on the network's vulnerability rather than protecting the complete assets. We can ignore the assets of the network that are not part of the critical assessment.

The study attempts to develop a model of the entire network, including the topology and connectivity between different devices, to know the exploitation of the attacker.

The simulation for the attacks can be done using the model proposed in this study, and the prediction for the attack paths will be possible at the time of compromise for various assets, which are very critical at any point in time.

The core idea behind the study is to identify all the possible attack paths that are vulnerable and make them secure. Automating the old and traditionally available tools to secure the network is the core point behind the study. The hardening of the network can be done with the help of the analysis proposed in this paper. With its help, it becomes really easy to identify all the vulnerabilities and secure the entire network.

In a win-win situation, the use of attack graphs depending on the topology will be beneficial to reduce the impact of attacks. After all the attack paths are identified, it is very easy to place the sensors to detect or prevent intrusion attempts at the most critical vulnerabilities. It is also possible for the attacker to figure out another vulnerability path within the network for a likely attack. But using the prescribed model, all possible paths that are vulnerable can be protected from any type of attack or network exploit. Even if the attacker tries to exploit a position in the network he thinks is good, but with the help of the model proposed in this study, the position the attacker thinks is simplified will be able to catch him and block the access. So, all the possible positions are stored with the vulnerabilities with the help of placement of intrusion detection sensor.

3.3 Attack Graph from the Network

The attack graphs created inside this study lead to the identification of all the vulnerable paths that can be used for the hardening of the network by the placement of sensors at the desired locations. These locations can be variable and change from network to network. The creation of vulnerability assessment can be done with the help of various vulnerability scanner tools. However, the generation of the adjacency matrix and the mathematical calculation needs to be done with the help of different computer vulnerability scorer tools. The attack graph creation algorithm is shown below:

Algorithm – Attack Graph Creation

```
Generate Attack Path (Graph g, Goal)
Input: as Directed Acyclic Forward Reachable Graph
Output: Generated Attack Paths 4.
Initialize stack pointer sp=0.
do
 Find all exploit sets that satisfy the Goal.
```

```
8.  Enqueue Queuesp with the chosen exploit set.
9.  Chose one of the exploit set from Queuesp.
10. Push it at Stacksp and dqueue from Queuesp.
11. Update the Goal with chosen exploit set and go to step 7.
12. Update stack pointer as sp=sp+1.
13. while (all preconditions of exploit set belongs to initial condition)
14. Read the stack from the top and get the attack path.
15. do
16. if (Queuesp is empty)
17.   Delete Queuesp.
18.   Delete the stack top exploit set and update sp=sp-1.
19. if (sp==-1)
20.     Go to Step 9.
21.   end if
22. while (not found an non empty Queuesp)
23. go to Step 9.
24. END Generate Attack Path
```

The general steps which are included in the network model to identify the vulnerabilities comprise following:

- Identifying all the machines in the network which are under consideration for vulnerability analysis.
- Networks scan for all the machines, and the services to identify the interconnection of the machines.
- Identifying all the critical and less critical services which are vulnerable to attacks.
- Creation of a matrix comprising the machine-to-machine mapping based on which the clustering algorithm can be applied.
- Creation of the attack graph depicting all the vulnerable parts and modelling the exception or rejection of vulnerability path as per the mathematical analysis.

The figure below represents a part of the attack graph for the network used above as a sample.

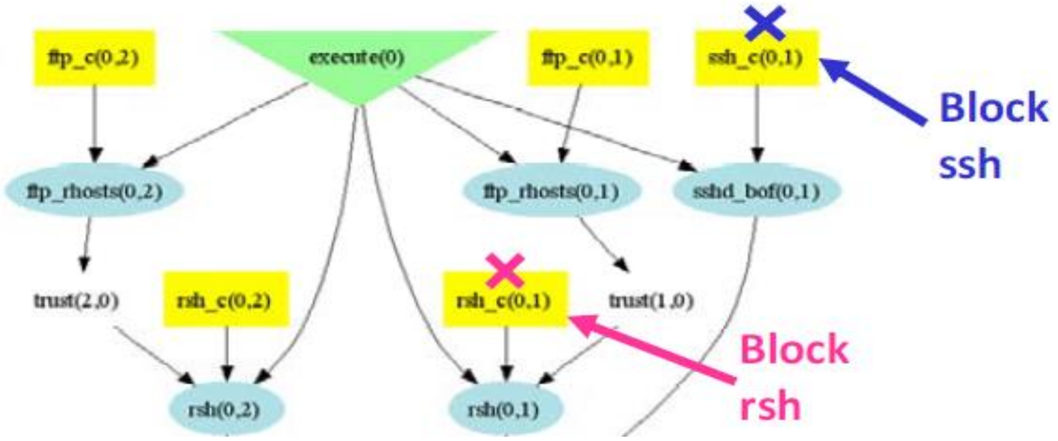

Figure 6: Attack Graph from the Network

Figure 6 depicts a network attack graph in which ssh_c (0, 1) represents the ssh service from computer 0 and computer terminal1.The edge represents connectivity between 0 and 1, and the number computer is extremely vulnerable to ssh attacks. It is also worth mentioning that rsh_c (0, 1) represents the service propagating from the same computer in the Sample network. The rsh services are also vulnerable to exploitation and should be checked. The two variant blocks which are visible in the diagram need to be checked and sensor placement can be done at the computer (0, 1) path. The identification of all such paths which ~~is~~ are responsible for safeguarding the network is the target of the study. The represented attack graph is a single-step vulnerability assessment. However, a similar approach can be applied towards m- Step vulnerability assessment, which is explained above.

- Calculating all the possible paths from the attack graph will comprise all the points prone to vulnerability attacks in the topology.
- Identifying the block points which are responsible or can be a hazard is towards vulnerability in the entire graph.
- Identifying the points which are reasonable enough for placement of sensors for an alarming situation is at the time of vulnerable attacks.
- Calculation of the risk is involved with the help of the risk matrix. This can be calculated using all the risk factors and the attack graph in unison.

The study, which is focused on this research, leads to the foundation of a very strong mechanism that can help handle all the vulnerabilities that can arise in a network. It will be really helpful for the minimization of the exploitation and saving the network from inappropriate access.

4. Results and Discussion

The study represents a new model for the visualization and the prediction of multiple steps attack graphs in any network. The software vulnerabilities can be identified and handled with the help of vulnerability scanners inside the network hosts in association with the firewall and the attacker's exploitable software codes. A complex attack graph can be created, for M step prediction, as well. As identification of vulnerability in the topology. Tab. 1, shows the exploited services and vulnerabilities in the demonstrative network.

Table 1: Exploited Services and Vulnerabilities in the Demonstrative Network

Host	Services	Vulnerabilities	OS
Host1	WuFTPD, SSH, RSH	sshd buffer overflow, ftp.rhost overwrite	Linux
Host2	ProFTPD,RSH,XTERM, DATABASE,	ftp.rhost overwrite, local xterm buffer overflow	Linux

The study that is done in this study refers to all the possibilities in which cyber vulnerabilities can be identified and the attacker can try to discover all the attack paths in a network. However, all the results may not be verified with the help of the demonstrator model. A thorough, in-depth analysis is required to mitigate all the potential risks and the attacks that are possible. Tab. 2, shows the cross-mapping between machines in the demonstrative network.

Table 2: Cross -Mapping between Machines in the Demonstrative Network

Relation	Host0	Host1	Host2
Host0	Localhost	FTP, SSH	FTP
Host1	Any	Localhost	FTP
Host2	any	FTP	Localhost

The attack graphs shown in Fig.6, lead to identifying all the vulnerable paths used to harden the network by the placement of sensors at the desired locations. These locations can be variable and change from network to network. The creation of vulnerability assessment can be done with the help of various vulnerability scanner tools. However, the generation of the adjacency matrix and the mathematical calculation needs to be done with the help of different computer vulnerability scorer tools. All the vulnerability paths that are identified as critical should be placed with sensors to isolate the intrusion. The problem stated in this particular research comprises many paths as the size of the network increases. It can be assumed to be an NP-Hard problem. We can simulate a similar problem regarding the set cover problem. And, in this case, we proceed with the greedy approach to determining an optimal network path rather than calculating all of the vulnerabilities at all of the paths, which may take more time and result in the integration of additional hardware in the form of sensors. Identified Attack Paths in the study shows the following problems and placement of the sensors can be done based on the paths observed.

Attack Path 1: {ftp_rhosts(0,2)}→{rlogin(0,2)}→ {local_bof(2,2)}

Attack Path 2: {sshd_ bof(0,1)}→{ f tp _ rhosts(1,2)}→{rlogin(1,2)}→ {local_bof(2,2)}

Attack Path 3:{ftp_rhosts{0,1)→{rlogin(0,1)}→{ftp_ rhosts(1,2)}→{rlogin(1,2)}→{local_ bof(2,2)}

The analysis and discovery of various critical services running on host machines. Priority of services running over hosts to avoid exploits shown in Fig. 7., and Path depth traversal increase in the vulnerability as shown in Fig. 8.

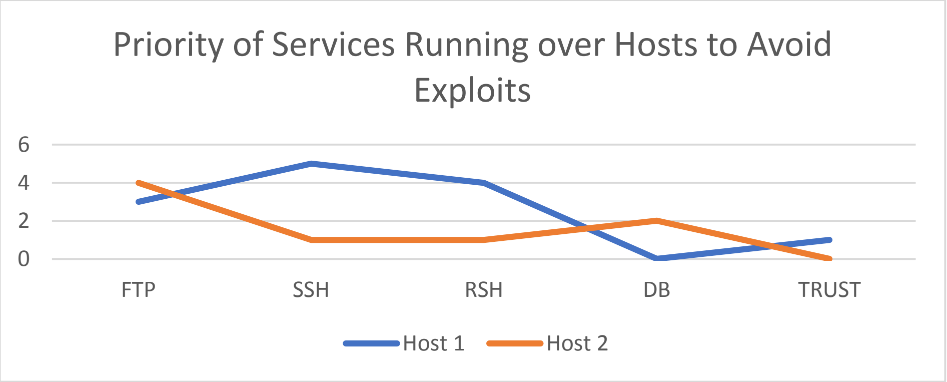

Figure 7: Priority-based Critical Processes

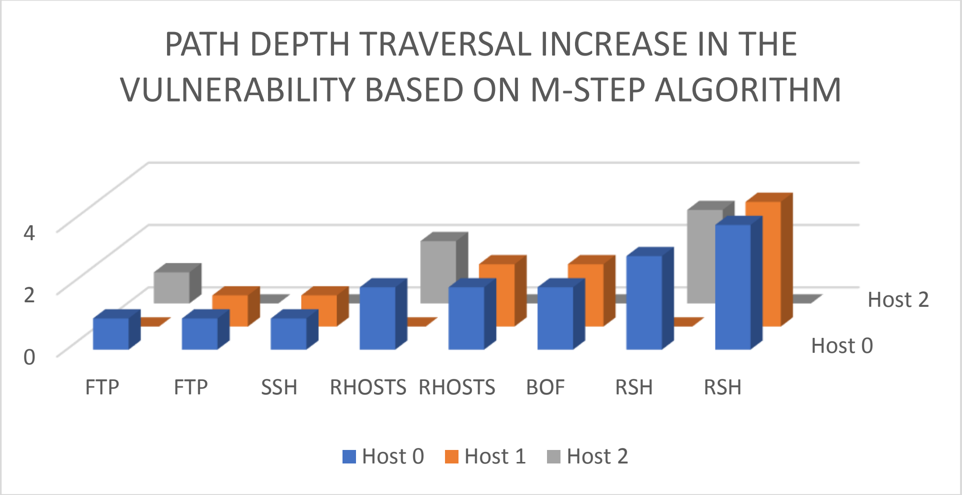

Figure 8: Traversal Depth in the graph.

5. Conclusion

The method presented in this study is a model that represents the network as part of the attack graph, which consists of different types of vulnerabilities other than services. After identifying all the services in this regard, the computing machines can be secured with the help of placing certain sensors and the system can be considered to have more security against the exploit. The proposed method visualizes and predicts complex multi-stage cyber-attacks. Traditional graph-centric modeling and adjacency matrix are used. The prediction for the vulnerabilities in the network can be found using this methodology. The principal analysis combines the important point that the attacker will always attack a site with many vulnerabilities for exploitation. This approach is continued with an enhanced version of network hardening, which identifies the areas where exploitation is possible and implements prediction parameters that include certain sensor-based studies. It makes correlation and prediction very simple to secure the network best. The attack graphs can be created and studied for further placement of vulnerability assessment tools. However, the geographical assessment can be done with the help of resources and powerful computing units. The study presented a model in this research that leads to a robust vulnerability analysis based on the network's topology. The creation of the attack graph makes it easy to identify grey areas and the research points. Thus, it is conducive to ensure cybersecurity and manage the network with the help of M-steps analysis.

6. Acknowledgment: The authors sincerely acknowledge the support from Majmaah University, Saudi Arabia for this research.

7. Funding Statement: The authors would like to thank the Deanship of Scientific Research at Majmaah University for supporting this work under Project Number No -R-14xx-4x.

8. Conflicts of Interest: The authors declare no conflict of interest.

9. References

Mishra. S., Sharma. S.K. and Alowaidi. M.A. (2021). Multilayer self-defense system to protect enterprise cloud," *CMC-Computer Materials & Continua,* vol. 66, no. 1, pp. 71-85.

Gupta. R., Tanwar. S., Tyagi. S., and Kumar. N. (2020). Machine learning models for secure data analytics: a taxonomy and threat model," *Computer Communications*, vol. 153, pp. 406-440.

Mishra. S., Sharma. S.K., and Alowaidi. M.A. (2020). Analysis of security issues of cloud-based web applications,". *Journal of Ambient Intelligence and Humanized Computing*, pp.1-12.

Mishra. S., and Alowaidi. M.A., Sharma. S.K. (2021). Impact of security standards and policies on the credibility of e-government,". *Journal of Ambient Intelligence and Humanized Computing*, pp.1-12.

Sarker. I.H., Abushark. Y.B., Alsolami. F. and Khan. A.I. (2020). Intrudtree: a machine learning based cyber security intrusion detection model," *Symmetry*, vol.12, no.5, pp.1-15.

Mishra. S. (2020). SDN-based secure architecture for IoT," *International Journal of Knowledge and Systems Science (IJKSS)*, vol.11, no,4, pp. 1-16.

Aldweesh. A., Derhab. A., and Emam. A.Z. (2020). Deep learning approaches for anomaly-based intrusion detection systems: A survey, taxonomy, and open issues," *Knowledge-Based Systems*, vol.189, pp.1-19.

Husak. M., Komarkova. J., Harb. E. B., and Celeda. P. (2018). Survey of attack projection, prediction, and forecasting in cyber security," *IEEE Communications Surveys & Tutorials*, vol.21, no.1, pp.640-660.

Liu. J., Lu. H., Wang. M. Chen J., and Zhang. Y. (2020). Macro perspective research on transportation safety: an empirical analysis of network characteristics and vulnerability," *Sustainability*, vol.12,*no*.15, pp.1-17.

Ghadi. M., Sali. A., Szalay. Z., and Torok. A. (2020). A new methodology for analyzing vehicle network topologies for critical hacking," *Journal of Ambient Intelligence and Humanized Computing*, pp.1-12.

Liu. S., Yu. Y., Hu. W., Peng. Y. and Yang. X. (2020). Intelligent vulnerability analysis for connectivity and critical-area integrity in IoV." *IEEE Access*, vol.*8*, pp.114239-114248.

Lallie. H.S., Debattista. K. and Bal. J., (2020). "A review of attack graph and attack tree visual syntax in cyber security," *Computer Science Review*, vol.35, pp.1-41.

Pirani. M., Taylor. J.A. and Sinopoli. B. (2021). "Strategic sensor placement on graphs," *Systems & Control Letters*, vol.148, pp.1-8.

Cinque. M., Della. C. and Pecchia. A. (2020). "Contextual filtering and prioritization of computer application logs for security situational awareness," *Future Generation Computer Systems*, vol.111, pp.668-680.

Stergiopoulos. G., Dedousi. P. and Gritzalis. D. (2021). "Automatic analysis of attack graphs for risk mitigation and prioritization on large-scale and complex networks in Industry 4.0," *International Journal of Information Security*, pp.1-23.

Pourhabibi. T., Ong. K.L., Kam. B.H. and Boo. Y.L. (2020). "Fraud detection: a systematic literature review of graph-based anomaly detection approaches," *Decision Support Systems*, vol.133, pp.1-15.

Ibrahim. M., Qays. A., Elhafiz. R., Alsheikh. A. and Alquq. O. (2020). "Attack graph implementation and visualization for cyber physical systems,"*Processes* vol.8, no. 1 ,pp.12.

Sansavini. F. and Parigi. V. (2020). "Continuous variables graph states shaped as complex networks: optimization and manipulation," *Entropy*, vol.22, no.1, pp.1-14.

Singhal. A. and Ou. X. (2017). “Security risk analysis of enterprise networks using probabilistic attack graphs,” *Network Security Metrics*, Springer, pp. 53-73.

Yang. S., Weirong. C., Xuexia. Z., Chenguang. L., Haifeng. W., Cui. W. *et al.*, (2019). “A graph-based model for transmission network vulnerability analysis,” *IEEE Systems Journal*, vol.14, no. 1, pp. 1447-1456.

Chen. X., Lau. N., and Jin. R. (2021). “PRIME: a personalized recommender system for information visualization methods via extended matrix completion,” *ACM Transactions on Interactive Intelligent Systems*, vol.11, no.1, pp.1-30.

Ramadan. R.A. (2020). “Efficient intrusion detection algorithms for smart cities-based wireless sensing technologies,” *Journal of Sensor and Actuator Networks*, vol.9, no.3, pp.1-22.

Russo. P., Caponi. A., Leuti. M. and Bianchi. G. (2019). “A web platform for integrated vulnerability assessment and cyber risk management,” *Information*, vol.10, no.7, pp.1-17

Ivanov. D., Kalinin. M., Krundyshev. V. and Orel. E. (2020). “Automatic security management of smart infrastructures using attack graph and risk analysis,” In *2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4)* IEEE, pp. 295-300.

Medvedev. D., Shani. U. and Dori. D. (2021). “Gaining insights into conceptual models: a graph-theoretic querying approach,” *Applied Sciences*, vol. 11,no.2, pp.1-29.

Nia. M.A., Bahrak. B., Kargahi. M. and Fabian. B. (2019). “Detecting new generations of threats using attribute-based attack graphs,” *IET Information Security*, vol.13,no.4, pp.293-303.

Das. S., Gregory. P., Lee. S, . Mehta. D and Suri. R. (2021). “Cybersecurity: the need for data and patient safety with cardiac implantable electronic devices,” *Heart rhythm*, vol.18, no. 3, pp. 473-481.

Ghazo. A. T., Ibrahim. M., Ren. H. and Kumar. R. (2019). “A2G2V: automatic attack graph generation and visualization and its applications to computer and SCADA networks,”. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no.10, pp.3488-3498.

C.Ma, Q. Lin, Y.Lin and X. Ma, (2021). “Identification of multi-layer networks community by fusing nonnegative matrix factorization and topological structural information,” *Knowledge-*

Based Systems, vol.213, pp.1-14

Hu. Z., Feiping. N., Chang. W., Shuzheng. H., Wang. R., Xuelong. L. *et al.*,(2020). "Multi-view spectral clustering via sparse graph learning," *Neurocomputing*, vol.384, pp.1-10.

Liu. L., Luo. S., Guo. F. and Tan. S. (2020). "Multi-point shortest path planning based on an improved discrete bat algorithm," *Applied Soft Computing*, vol. 95, pp.1-10.

Chen. L., Yue. D., Dou. C., Chen. J. and Cheng .Z. (2020). "Study on attack paths of cyber-attack in cyber-physical power systems, *IET Generation, Transmission & Distribution*, vol.14, no.12, pp.2352-2360.

Doi: https://doi.org/10.52132/Ajrsp.e.2022.33.3