Propose Vulnerability Metrics to Measure Network Secure using Attack Graph

Zaid. J. Al-Araji[1], Sharifah Sakinah Syed Ahmad[2], Raihana Syahirah Abdullah[3]
Faculty of Information Communication Technology
Universiti Teknikal Malaysia, Melaka
Melaka, Malaysia

***Abstract*—With the increase in using computer networking, the security risk has also increased. To protect the network from attacks, attack graph has been used to analyze the vulnerabilies of the network. However, properly securing networks requires quantifying the level of security offered by these actions, as you cannot enhance what you cannot measure. Security metrics provide a qualitative and quantitative representation of a system's or network's security level. However, using existing security metrics can lead to misleading results. This work proposed three metrics, which is the Number of Vulnerabilities (NV), Mean Vulnerabilities on Path (MVoP), and the Weakest Path (WP). The experiment of this work used two networks to test the metrics. The results show the effect of these metrics on finding the weaknesses of the network that the attacker may use.**

Keywords—Attack graph; security metrics; attack path; path analysis; attack graph uses

I. Introduction

Nowadays, the use of network technology has increased [1], [2]. Nonetheless, since the network is advantageous for people to live and work in, it also carries security problems that must not be overlooked [2]. In many key computer systems and applications, security has been and will remain a major concern. A comprehensive cybersecurity attack will significantly harm the target system as well as the credibility of the businesses or organizations that use it [3]. An attacker may use such attacks to get access to private data, degrade network performance, and eventually take complete control of the targeted system. To detect or protect the network from attacks, the researchers have used many methods [4]. One of these methods in vulnerabilities analysis is an attack graph.

Attack graph has been used for the first time by Philips and Swiler [5], [6]. Since then, researchers have suggested many methods to produce an attack graph. For example, Ammann et al. (2002) proposed the generation method based on monotonicity [7], while Vaibhav Mehta et al. (2006) proposed a ranking attack graph relying on graph neural network (GNN) [8]. Furthermore, Apart from that, Yun Chen et al. (2017) proposed an attack graph generation algorithm relying on a supervised Kohonen neural network [9], while HengLi et al. (2017) introduced a searching forward complete attack graph generation algorithm depending on hypergraph partitioning [10]. Also, Bintao Yuan et al. (2020) introduced the network vulnerability assessment method depending on the graph database and elaborated its efficacy in solving state explosion and other methods [11].

An attack graph may be utilized for many reasons, with positive or negative consequences [13]. Typically, attack graphs are used by researchers to improve the network's security. One of these applications is the computation of network security metrics. Attack graphs may be employed to generate network security metrics to analyze the target network's overall security. These metrics may be utilized to assess the target network's security risk. National Institute of Standards and Technology (NIST) describes security metrics as techniques that gather, analyze, and report pertinent performance-related data to aid decision-making, maximize performance, and increase transparency [12].

There are many security metrics proposed by researchers, such as Shortest Path (SP) Metric, Mean of Paths Length (MoPL) Metric, Number of Paths (NP) Metrics, etc. Some are combined to get new metric with new features and better results, like combining NP and MoPL to get Standard Deviation of Path Lengths (SDPL) Metric proposed by [13].

In[14], the authors divided the attack graph-based security metrics into two types, which are host and network-based metrics. Host-based security measures the level of security of individual hosts in a network. The host-based is divided into two types, which are with and without probability. Meanwhile, the network-based uses the structure of a network to aggregate the network's security property. This type of metrics is classified into two categories, which is path and non-path metrics.

However, using these metrics sometimes gives misleading results, failing to sufficiently account for the number of ways an attacker violates a security policy. In this case, not only the number of the ways but the accuracy is also responsible. For example, the shortest path is not necessary to be the path used by the attacker. It also does not take into account the attack effort connected to the attack paths.

In this paper, three metrics are proposed, which are Number of Vulnerability (NV) Metric, Mean Vulnerabilities on Path (MVoP) Metric, and Weakest Path (WP) Metric to reduce the misleading of the security metrics. The NV and MVoP will view how strong the network is and indicate how much effort the attacker needs to breach network security. On the other hand, it will also measure how much effort is required by the administrator to guard the network from any attacker. Meanwhile, the WP metric views the network's weakest path, which allows the attacker to breach the network policy with minimum effort.

The remainder of the paper is laid out as follows. Section 2 presents the attack graph overview, while Section 3 gives the security metrics related work, Section 4 proposes the security metrics, Section 5 is the experiment performed and results, while Section 6 gives the conclusion.

II. Attack Graph Background

The concept of attack graph has been proposed by Philips and Swiler [15], As shown in Fig. 1. Since then, many researchers have generated attack graphs differently using different methods to improve the attack graph. It is a security model denoting the chains of vulnerabilities, where exploits in the network can be in various forms. The attack graph representation can be organized [16] as a state-oriented, exploit-oriented, or state-exploit-oriented attack graphs [17]. Attack graph generation helps merge low-level vulnerabilities to display all attack paths from source to network goals. By examining the exploited attack paths, security experts should concentrate on patches or configuration bugs that present greater risks. The probabilistic attack graph's risk assessments support such decisions even more [18].

Attack graph generation has three steps which are reachability, attack model, and core building. The attack graph reachability explores the conditions of accessibility in the network, defining whether two given devices could reach one another. The most common representation of network reachability data is a reachability matrix, in which the rows and columns represent the network's hosts. Moreover, each entry indicates the reachability condition between the hosts on the corresponding row and column, respectively [20]. Various connections between the hosts may be represented by a reachability matrix, including transport, network, physical, and application-level connections. Its spatial complexity is on the order of the square of the network's number of hosts [20].

The second phase is the attack model. Attack graph modelling deals with the modelling of attack templates, determining attack graph structure, and modeling networks. The attack template modelling comprises the representation of pre- and post-conditions for the vulnerability. It also provides a process by which information in public vulnerability and weakness databases can be extracted from these conditions for particular vulnerabilities [20].

The attack graph structure's determination involves determining which node and edge types could be contained in the attack graph. Network modelling aims to define a suitable representation of network information [6]. The third phase is the attack graph core building, which denotes the main algorithm employed to develop the attack graphs. Many paths will be pruned in this stage during creating the resulting attack graph in this process [20]. From two different viewpoints, an attack graph core building mechanism could be taken into account. One is the method of evaluating the attack paths, and the other is the method of pruning the attack paths [20].

Generating an attack graph may be utilized for various reasons comprising negative or positive impacts. According to [20], attack graph can be used in four prespectives as in Fig. 2.

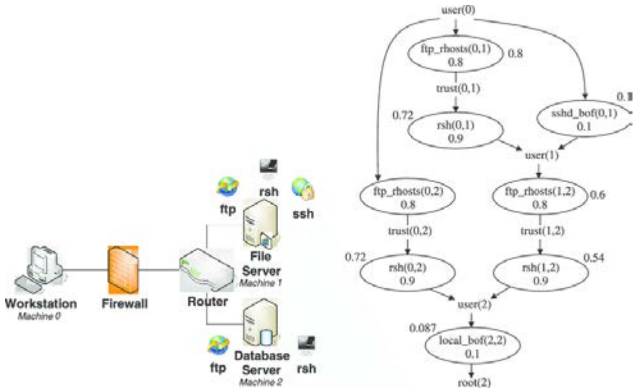


Fig. 1. An Attack Graph Sample [19].

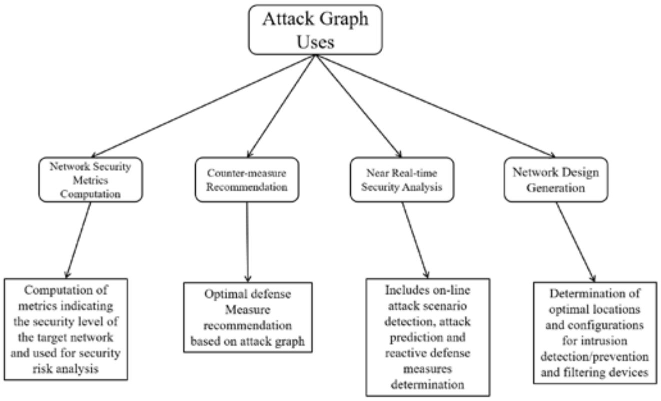

Attack Graph Uses
Network Security Metrics Computation
Counter-measure Recommendation
Near Real-time Security Analysis
Network Design Generation
Computation of metrics indicating the security level of the target network and used for security risk analysis
Optimal defense Measure recommendation based on attack graph
Includes on-line attack scenario detection, attack prediction and reactive defense measures determination
Determination of optimal locations and configurations for intrusion detection/prevention and filtering devices

Fig. 2. Attack Graph uses [20].

Attack graphs can be used for recommending near optimal security defense counter-measures. Optimal counter-measure recommendation can also find practical usage for determining proactive defense recommendations. It can also use the attack graphs generated by accounting for the goal privileges pointing to critical network resources [21].

Another use for the attack graph is network design generation can find practical use in locating the intrusion detection/prevention systems and firewalls optimally in the target network. It can also be used to determine firewall and access control rules, if necessary support for resolving conflicting rules and processing different custom rule formats is provided [17].

Attack graphs can be used for on-line security situational assessment (monitoring) and detecting ongoing attack scenarios by performing highlevel correlation and aggregation of the intrusion alerts and system logs collected throughout the target network. The detected attack scenarios can be used to perform future attack predictions and determine reactive defense measures [17].

Attack graphs also can be used to derive network security metrics used for global security assessment of the target network. These metrics can be used to perform security risk analysis for the target network. Each node (generally indicating a network state) and each edge (generally indicating a vulnerability exploit) on the attack graph can be assigned a probability of occurrence. A node can also be assigned a possible damage value, if the corresponding network state for the node indicates the compromise of some information source for a network host. From these probability and damage values, the cumulative risk values are computed for each network state on the attack graph [17]. In this paper, we will use the attack graph to derive the security metrics.

III. Security Metrics

Metrics, as defined by the NIST, are instruments that gather, analyze, and report applicable performance-related data to aid decision-making and increase performance and transparency. Comprehensive network security and CSA management necessitate the use of security metrics [12].

Security metrics have different categories. Based on Nwokedi C. Idika [22], security metric can be classified into two main classes, which are primary and secondary as in Fig. 3. The primary security metric classes are architectural-based security metrics and performance-based security metrics. The difference in the two classes stems from the type of attributes they measure. Architectural-based metrics measure internal attributes. Performance-based metrics measures external attributes. The secondary security metric classes are security metrics, complexity-based security metrics and time-based security metrics. These metrics can be applied to internal and external attributes of a network. Most of the primary class belong to secondary class as well but not all metrics belong to a primary class but not necessarily a secondary class.

Attack graph-based security metrics is a type of architectural metric [22]. It is a value produced from measuring the internal attributes of a network that affect IT security or operational security. The values are derived from generating an attack graph and subsequently deploying an analysis over the attack graph. This analysis is the measurement that produces the attack graph-based security metric [22].

According to Enoch et al. (2017) attack graph security metrics can be divided to two categories depending on the network reachability which are Host-based and Network-based as in Fig. 4 [23].

The host-level metrics are used to quantify the security level of individual hosts in a network. The host based metrics are classified to two categories which are security metrics with probability and security metrics without probability [24]. The classification had been done because sometimes it is infeasible to find a probability value for an attack, and some analysis and optimisation can be done with or without probability assignments [20].

The network-level metrics are used the structure of a network to aggregate the security property of the network. The network-based security can be classified to two categories which are security metrics path-based and security metrics non path-based. Path based metrics use the reachability information of a network to quantify the security level of the network. While in non path-based metrics, the structure and attributes of a network are not considered; instead, the security of a network is quantified regardless of the network structure [22]. Researchers had proposed many metrics. In this section, some of the previous works will be explained.

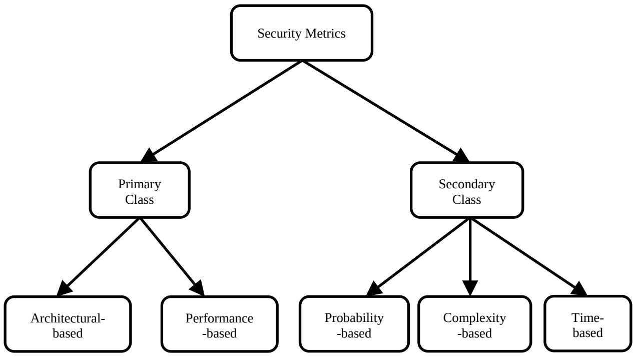


Fig. 3. Security Metrics Classification [22].

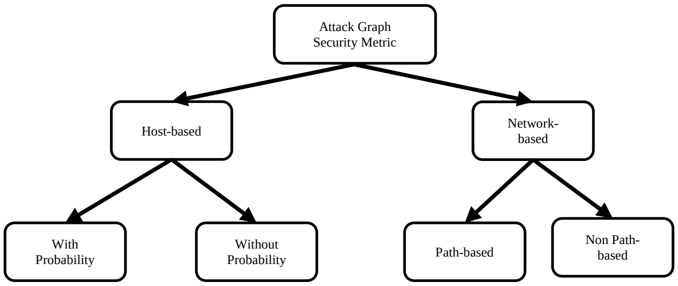


Fig. 4. Attack Graph Security Metrics Classification [23].

Phillips and Swiler (1998) proposed the Shortest Path (SP) Metric. The shortest attack path is the one that takes an attacker from his initial state to his desired goal state using the shortest distance. The length function used to calculate the distance is determined by the security engineer who conducts the attack graph analysis. Nonetheless, this metric does not signify the number of shortest paths in a network. Also, there is no guarantee that the short path is the path used by the attacker [5]. Because of the metric is depending of the length of the path, this metric is path-based metric.

Moreover, Ortalo (1999) suggested the Number of Paths metric (NP). The number of attack paths in a given attack graph is expressed by this security metric. It measures how vulnerable a network is to be attacked. A larger Number of Paths metric suggests a more exposed network. This metric, however, does not account for attack effort, implying that two networks with the same number of attack paths are considered to be of equal security [25]. This metric is path metric because it counts the network paths.

Furthermore, Idika details the Mean of Path Lengths metric (MPL), first introduced by Wei Li (2006) as the Average Path Length metric [26]. It calculates the arithmetic mean of all path lengths to reflect the typical path length. It also estimates how much effort an attacker would impose to break a network security policy. Since an attacker may not have the same perspective of known vulnerabilities as a security engineer, this metric is important. Because of this lack of experience, the attacker may pick a path that is not the shortest. Alternatively, an attacker may choose the other path because the attacker believes the security engineer is using the shortest path analysis. However, this metric cannot be applied alone because it depends on the NP metric [13]. This metric is considered as path metric because it calculate the average of the path length of the network.

These security metrics can be used to retrieve security-relevant data, but they can also produce false results. The Shortest Path and Mean of Path Lengths metrics do not account for all the possible ways an attacker would break a security policy. The attack effort related to the attack paths is not fully accounted for by the Number of Paths metric. To overcome these problems, this work proposes three metrics, which is NV, MVoP, and WP explained in the next section.

IV. Proposed Metrics

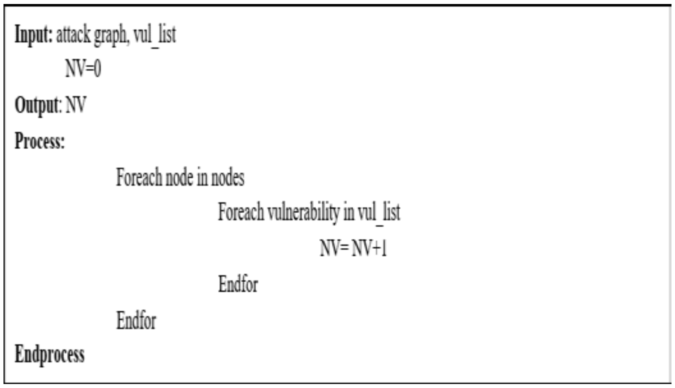
In this section, three attack graph-based security metrics will be proposed: Number of Vulnerabilities Metric (NV), Mean of Vulnerabilities on Path Metric (MVoP), and Weakest Path Metric (WP).

A. Number of Vulnerabilities (NV) Metric

The Number of Vulnerabilities (NV) Metric represents the number of weakness in each node of the network that an attacker can use to cross privilege boundaries in the network. This metric aims to understand the number of disadvantages in the network and allow the administrator to fix it and compare the security of two networks with different size and topology. The formalization of the NV metric is presented in equation 1:

$$NV = \sum V(p_1, p_2, \dots, p_n) \tag{1}$$

Here, V represents the vulnerabilities, p represents the path, and n denotes the number of nodes. Thus, the metrics will calculate the vulnerabilities for each path starting from p_1 to p_n. The pseudocode of NV metric calculation is in Fig. 5. Basically the input in the pseudocode is the attack graph to select all the nodes and the vul_list which represent the vulnerabilities list. The process is so simple is to select a node from all the nodes in the attack graph and calculate the number of vulnerabilities in that node and add them to the counter. This metric is host-based and without probability metric because this metric calculate the vulnerabilities from the host.



```
Input: attack graph, vul_list
        NV=0
Output: NV
Process:
                Foreach node in nodes
                        Foreach vulnerability in vul_list
                                NV= NV+1
                        Endfor
                Endfor
Endprocess
```

Fig. 5. NV Metric Calculation.

B. Mean Vulnerabilities on Path (MVoP) Metrics

The Mean Vulnerabilities on Path (MVoP) metric represents the average number of the path's vulnerabilities. This metric indicates how much effort an attacker would have to put in to break a network security policy. It also provides a view for the defender to expect the attacker's move. The formalization of the NV metric is presented in equation 2:

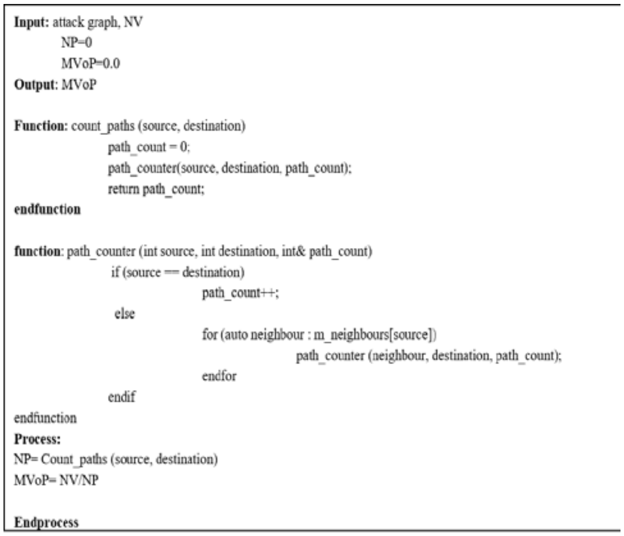
$$MVoP = \frac{NV}{NP} \tag{2}$$

where NV implies the number of vulnerabilities on the network, while NP is the number of the network paths. The formalization of NP is presented in equation 3:

$$NP = |p_1, p_2, \dots, p_k| \tag{3}$$

where p represents the path and k represents the number of the path. Fig. 6 shows the NP and MVoP metrics calculation. To calculate MVoP metric, we need to calculate NV and NP metrics. The NV metric has been calculated above (see Section 4.1). In this section, we will calculate the NP metric.

The calculation of NP metric is depending on the edges between the nodes, basically each node has many edges with other nodes, to calculate this edges, we used edges list for each node, the we start counting the path from the source to destination using edge list. During the calculation the source change depending on the edge list until the source equal the destination which is mean it is a path. After calculating the numbe of the path, we calculate the MVoP by dividing NV on NP. This metric is considered as path-based and without probability metric because it calculate the vulnerabilies number from the host and calculate the path number from the network.



```
Input: attack graph, NV
        NP=0
        MVoP=0.0
Output: MVoP

Function: count_paths (source, destination)
                path_count = 0;
                path_counter(source, destination, path_count);
                return path_count;
endfunction

function: path_counter (int source, int destination, int& path_count)
                if (source == destination)
                        path_count++;
                else
                        for (auto neighbour : m_neighbours[source])
                                path_counter (neighbour, destination, path_count);
                        endfor
                endif
endfunction
Process:
NP= Count_paths (source, destination)
MVoP= NV/NP

Endprocess
```

Fig. 6. MVoP Metric Calculation.

C. Weakest Path (WP) Metric

The Weakest Path (WP) Metric is similar to the shortest path metric but in another term, which is the network's strength. The path's strength does not depend on the NV only but also on the score of vulnerability itself. The vulnerability score can use the CVSS that NIST had invented. The formalization of calculating the path score using CVSS is presented in equation 4:

$$V(p) = \frac{\sum_{k=1}^{k=n} CVSS(v(a,b))}{n}, \tag{4}$$

where p represents the path, v represents the vulnerability between node a and b, k denotes the number of the nodes in the path, and n refers to the number of the nodes. Basically, the equation calculates the average score of each vulnerability in the path.

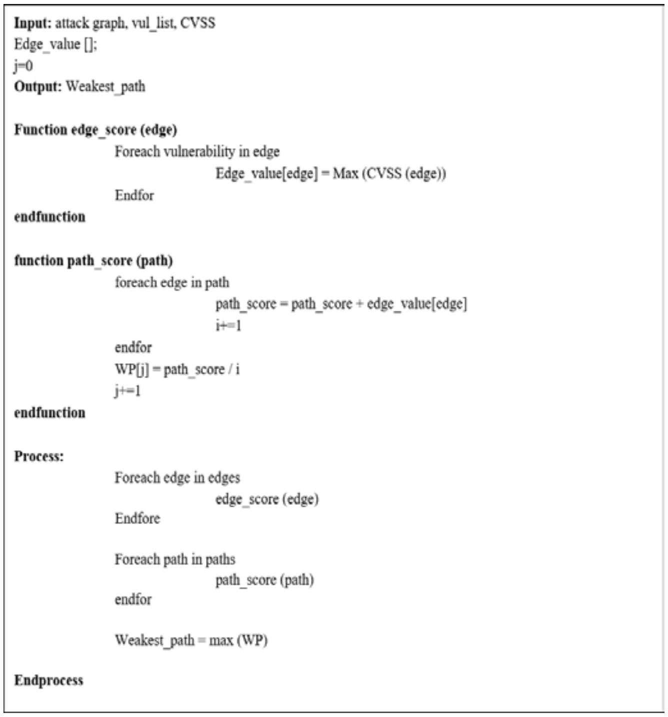
After calculating the path vulnerability score, the score results are compared between all paths in the network to represent the weakest path. The formalization of the WP metric is defined in equation 5:

$$WP = \max\left(V(p_1), V(p_2), \dots, V(p_k)\right) \quad (5)$$

where p represents the path, k represents the path number, and $V(p_k)$ represents the summation of the path vulnerabilities score. The calculation of the WP metric in Fig. 7.

The calculation of the weakest path is depending on the value of each edge in the path. To calculate the edge value, we need to find the vulnerabilities score of the edge, so the input will be the attack graph, vulnerabilities list and vulnerability score (CVSS). Then we calculate the edge score by finding the maximum vulnerability score in edge.

After calculating the edge value, the path score will be calculated. Basically, the value of the path will be the average number of edge value in the path. The path score will be saved to compare it with other paths score to find the weakest. The highest path score in the paths will be the weakest path in the network. This metric is considered as path-based metric because it calculate the path score.



```
Input: attack graph, vul_list, CVSS
Edge_value [];
j=0
Output: Weakest_path

Function edge_score (edge)
        Foreach vulnerability in edge
                Edge_value[edge] = Max (CVSS (edge))
        Endfor
endfunction

function path_score (path)
        foreach edge in path
                path_score = path_score + edge_value[edge]
                i+=1
        endfor
        WP[j] = path_score / i
        j+=1
endfunction

Process:
        Foreach edge in edges
                edge_score (edge)
        Endfore

        Foreach path in paths
                path_score (path)
        endfor

        Weakest_path = max (WP)

Endprocess
```

Fig. 7. WP Metric Calculation.

V. Experiment and Results

In this section, the experiment of the three metrics had been done. The experiment used two attack graphs generated with two different networks. In the following, the network's topology will be explained. Then, the results of applying the metrics will be discussed.

A. Network Design

In this experiment, we chose two networks to test the effect of our proposed metrics. The network A has two workstations, three servers, a firewall and an external attacker, as in Fig. 8. The network connectivity in this network topology depends on firewall rules given as follows:

- There is bidirectional connectivity between the webserver and other machines in the network.
- The external host is the attacker located on the internet and has access to the webserver through HTTP protocol and HTTP port.
- The two workstations and fileserver have access to each other through NFS protocol and NFS port.
- The two workstations and fileserver have access to the internet through HTTP protocol and HTTP port.
- The two workstations have access to the database server.

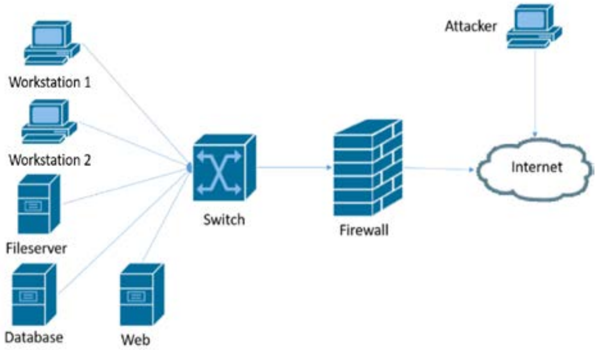


Fig. 8. Network a Topology.

Meanwhile, the network B has two workstations: a web server and database server and an attacker, as in Fig. 9. The network connectivity in this network topology depends on firewall rules listed as follows:

- There is bidirectional connectivity between the webserver and other machines in the network.
- The external host is the attacker located on the internet and has access to the webserver through HTTP protocol and HTTP port.
- Workstation 4 has access to the database server.
- All workstations have access to the internet through HTTP protocol and HTTP port.
- All workstation has a connection with each other.

The experiments evaluated the generating attack graph in the following environment. The CPU is core i5 2.0 GHz with 8 GB of RAM, the operating system is windows 10, and the coding was performed using Microsoft Visual Studio C# 2012. Also, the Common Vulnerabilities and Exposures (CVE) and National Vulnerability Database (NVD) lists were employed, which were provided by NIST to load the vulnerabilities to the attack graph.

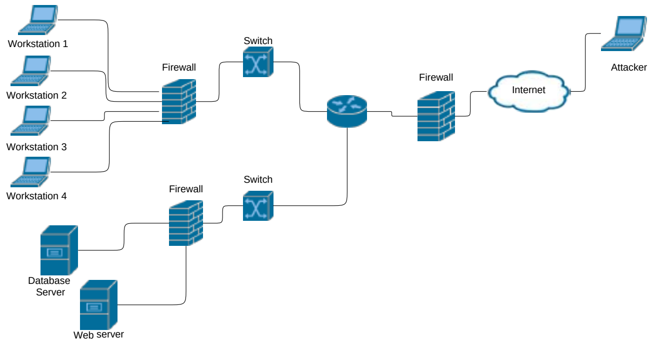


Fig. 9. Network B Topology.

B. Results

In this section, the findings of the experiment will be discussed. Before explaining the results, the paths from the started node (the attacker) to the targeted node (database server) in both networks and list of vulnerabilities must be identified.

The number of the paths in network A is four (4) paths, while the network B has 14 paths, as illustrated in Table I. The paths extracted from the networks based on the topology of the networks.

Here, A represents the word Attacker, W represents the word webserver, D represents the word database, and the numbers represent the workstations. The Attacker can reach the workstations through the webserver only. In this case the next step for the Attacker is webserver. While the webserver has bidirectional connection to the workstations, but the webserver cannot reach directly to the database server which only can be reached by both workstations in network A and workstation 4 in network B, in this case the Attacker need to conquest the workstations before can conquest the database server by exploit the vulnerabilities in the network.

A vulnerability is identified by CVE as a weakness in the computational logic (e.g., code) found in software and hardware components. Usually, the node can have more than one vulnerability because it depends on the applications that installed and the hardware is used. This section will explain some of these vulnerabilities that found in network A and network B as in Table II.

The first vulnerability that discovered is CVE-2010-0490, this weakness in Internet Explorer 6, 7 and 8 with the possibility that remote intruder can execute arbitrary code on the target machine. While CVE-2014-2510 in the outlook, this vulnerability remote authenticated users to read arbitrary files via an external entity declaration in conjunction with an entity reference. The next vulnerability is CVE-2018-15983 in adobe flash player, this vulnerability has an insecure library loading (dll hijacking) vulnerability. Successful exploitation could lead to privilege escalation. The CVE-2010-0483 vulnerability discovered in VBScript, this vulnerability might allow user-assisted remote attackers to execute arbitrary code via a long string in the fourth argument (aka helpfile argument) to the MsgBox function.

1) *NV and MVoP:* Upon implementing the NV metric on both graphs, the results show that the network A has more vulnerabilities than the network B, as displayed in Table IV. Even the network B has more nodes than the network A, but it has fewer vulnerabilities than the network A. The reason is that the nodes can have more than one vulnerabilities at the same time. Depending in this metric network B is more secure than network A, unlike NP metric which shows than network A is more secure than network B. The reason of the different between two metrics which is the NV metric is depending on the weaknesses of the network not like NP metric which depend on the paths only.

Also, for the MVoP metric, the network A has more vulnerabilities in each path than the network B. The reason for that is the network A has few paths and more vulnerabilities than the network B, as displayed in Table III. The result of MVoP metric shows that network B is more secure than network A, unlike MPL metric which shows network A is more secure. The reason is MVoP depends on the average number of the weaknesses in the path, while MPL depends on the average length of the paths.

TABLE I. PATHS OF THE NETWORKS

No	Network A paths	Network B paths
1	A-W-1-D	A-W-4-D
2	A-W-2-D	A-W-1-4-D
3	A-w-1-2-D	A-W-2-4-D
4	A-W-2-1-D	A-W-3-4-D
5	-	A-W-1-2-4-D
6	-	A-W-1-3-4-D
7	-	A-W-2-1-4-D
8	-	A-W-2-3-4-D
9	-	A-W-3-1-4-D
10	-	A-W-3-2-4-D
11	-	A-W-1-2-3-4-D
12	-	A-W-1-3-2-4-D
13	-	A-W-2-1-3-4-D
14	-	A-W-2-3-1-4-D
15	-	A-W-3-1-2-4-D
16	-	A-W-3-2-1-4-D

TABLE II. EXAMPLE VULNERABILITIES

No	Vulnerability	CVSS Score
1	CVE-2010-0490	9.3
2	CVE-2014-2510	6.8
3	CVE-2018-15983	7.8
4	CVE-2010-0483	7.6

TABLE III. METRICS IMPLEMENTATION RESULTS

Metrics	Network A	Network B
NV	18	15
MVoP	4.5	1.07
NP	4	16
MPL	3.5	5.06

The result of NP metric and MPL metric are not accurate enough of measuring the strength of the network because they do not see the detail of the network, while NV and MVoP metrics are more accurate because of calculating the weaknesses of the network.

2) WP metric: For the third metric, WP metric, the experiment compares the path's strength between the same network paths. To calculate the path strength, the metric takes the highest score vulnerability in the link between two nodes to determine the path's score. Table IV shows the score between all links in both networks.

The results of the edges strength had been calculating depends on the highest vulnerability score in the edges. There are many vulnerabilities had been dropped in the calculation because the score of these vulnerabilities is smaller than the vulnerabilities that been calculated because we assumed that the vulnerabilities that have higher score are more danger and have a high chance to be exploit by the attacker. Based on these edges scores, the weakest path had been calculated as in Table V.

Table VI shows that path number 2 in the network A and path number 8 in the network B has the highest number of WP metric calculation, implying the weakest path in the network.

Comparing with SP metric, the results show that WP metric are more specific, accurate and effective, as in Table VI.

TABLE IV. EDGE VULNERABILITY SCORE

No	Network A links score	Link Score	Network B links score	Link score
1	A-W, W-A	9.8	A-W, W-A	9.2
2	1-2, 2-1	7.8	1-2, 2-1	5.4
3	W-1, 1-W	6.2	1-3, 3-1	8.1
4	W-2, 2-W	8.1	1-4, 4-1	7.2
5	1-D, D-1	6.8	2-3, 3-2	8.1
6	2-D, D-2	8.8	2-4, 4-2	6.4
7			3-4, 4-3	7.8
8			W-1, 1-W	7.3
9			W-2, 2-W	7.8
10			W-3, 3-W	6.1
11			W-4, 4-W	6.4
12			D-4, 4-D	8.1

TABLE V. WP METRIC SCORE FOR BOTH NETWORKS

Number	Network A path	WP metric	Network B path	WP metric
1	A-W-1-D	7.6	A-W-4-D	7.9
2	A-W-2-D	8.9	A-W-1-4-D	7.95
3	A-W-1-2-D	8.15	A-W-2-4-D	7.87
4	A-W-2-1-D	8.12	A-W-3-4-D	7.8
5	-		A-W-1-2-4-D	7.28
6	-		A-W-1-3-4-D	8.1
7	-		A-W-2-1-4-D	7.5
8	-		A-W-2-3-4-D	8.2
9	-		A-W-3-1-4-D	7.74
10	-		A-W-3-2-4-D	7.58
11	-		A-W-1-2-3-4-D	7.65
12	-		A-W-1-3-2-4-D	7.86
13	-		A-W-2-1-3-4-D	7.73
14	-		A-W-2-3-1-4-D	8.08
15	-		A-W-3-1-2-4-D	7.21
16	-		A-W-3-2-1-4-D	7.35

TABLE VI. METRICS COMPARISON

Metric	Network A	Network B
WP	- A-W-2-D	A-W-2-3-4-D
SP	- A-W-1-D - A-W-2-D	A-W-4-D

The SP metric has two paths in the network A, causing misleading results, while the WP metric has been more specific and gets one path only. In the network B, the SP and WP metrics get different paths because the SP metric counts the node number between the started and targeted node. Simultaneously, the WP is more specific and calculates the vulnerability score in each link, giving the easiest path for the attacker to reach the target.

VI. CONCLUSION

In this paper, three metrics had been proposed, which is NV, MVoP, WP metrics. These three metrics depend on vulnerabilities as a major factor to measure the security of the network. The experiment had been performed in two attack graphs generated using two different networks. The results show that the network A has more vulnerabilities, while the MVoP is higher than the network B, even though it has more nodes and paths. For the last metric, the WP metric, the network's A result shows that the shortest path is the weakest path of the network while it was not the shortest path in the network B.

VII. Future Work

Further investigation and research are still required, especially in the flowing fields:

- The work developed using the metrics and the experiments will be performed for larger graphs.
- Also, we will attempt to combine the metrics to obtain better results.

References

[1] Ramos, M. Lazar, R. Holanda Filho, and J. J. P. C. Rodrigues, "Model-based quantitative network security metrics: A survey," IEEE Commun. Surv. Tutorials, vol. 19, no. 4, pp. 2704–2734, 2017.

[2] Z. J. Al-araji, S. S. A. Syed, M. W. Al-salihi, H. A. Al-lamy, M. Ahmed, and W. Raad, "Network Traffic Classification for Attack Detection Using Big Data Tools : A Review," Intell. Interact. Comput. Lect. Notes Networks Syst. 67, pp. 355–363, 2019, doi: 10.1007/978-981-13-6031-2.

[3] A. A. Hassan, W. M. Shah, M. F. I. Othman, and H. A. H. Hassan, "Evaluate the performance of K-Means and the fuzzy C-Means algorithms to formation balanced clusters in wireless sensor networks.," Int. J. Electr. \& Comput. Eng., vol. 10, no. 2, 2020.

[4] M. A. Mohammed et al., "A comprehensive investigation of machine learning feature extraction and classification methods for automated diagnosis of covid-19 based on x-ray images," Comput. Mater. Contin., vol. 66, no. 3, 2020.

[5] C. Phillips and L. P. Swiler, "A Graph-based System for Network-vulnerability Analysis," Proc. 1998 Work. New Secur. Paradig., pp. 71–79, 1998, doi: 10.1145/310889.310919.

[6] L. P. Swiler, C. Phillips, D. Ellis, and S. Chakerian, "Computer-attack graph generation tool," Proc. - DARPA Inf. Surviv. Conf. Expo. II, DISCEX 2001, vol. 2, pp. 307–321, 2001, doi: 10.1109/DISCEX.2001.932182.

[7] P. Ammann, D. Wijesekera, and S. Kaushik, "Scalable, graph-based network vulnerability analysis," Proc. ACM Conf. Comput. Commun. Secur., no. June, pp. 217–224, 2002, doi: 10.1145/586110.586140.

[8] V. Mehta, C. Bartzis, H. Zhu, E. Clarke, and J. Wing, "Ranking attack graphs," in International Workshop on Recent Advances in Intrusion Detection, 2006, pp. 127–144.

[9] Y. Chen, K. Lv, and C. Hu, "Optimal Attack Path Generation Based on Supervised Kohonen Neural Network," vol. 2, pp. 399–412, 2017, doi: 10.1007/978-3-319-64701-2.

[10] H. Li, Y. Wang, and Y. Cao, "Searching Forward Complete Attack Graph Generation Algorithm Based on Hypergraph Partitioning," Procedia Comput. Sci., vol. 107, no. Icict, pp. 27–38, 2017, doi: 10.1016/j.procs.2017.03.052.

[11] B. Yuan, Z. Pan, F. Shi, and Z. Li, "An Attack Path Generation Methods Based on Graph Database," in 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), 2020, vol. 1, pp. 1905–1910.

[12] Y. Cheng, J. Deng, J. Li, S. A. Deloach, and A. Singhal, Metrics of Security, vol. 62. Springer International Publishing Switzerland 2014, 2014.

[13] N. Idika and B. Bhargava, "Extending attack graph-based security metrics and aggregating their application," IEEE Trans. Dependable Secur. Comput., vol. 9, no. 1, pp. 75–85, 2012, doi: 10.1109/TDSC.2010.61.

[14] M. G. and D. S. K. Simon Enoch Yusuf, Jin B. Hong, "Composite Metrics for Network Security Analysis." Journal ofSoftware Networking, pp. 137–160, 2017.

[15] C. Phillips and L. P. Swiler, "A graph-based system for network-vulnerability analysis," in Proceedings of the 1998 workshop on New security paradigms, 1998, pp. 71–79.

[16] X. Ou, "A logic-programming approach to network security analysis," Ph.D Diss., no. November, 2005.

[17] K. Kaynar, "A taxonomy for attack graph generation and usage in network security," J. Inf. Secur. Appl., vol. 29, pp. 27–56, 2016, doi: 10.1016/j.jisa.2016.02.001.

[18] M. U. Aksu, M. H. Dilek, E. İ. Tatlı, K. Bicakci, and M. Ozbayoglu, "Automated Generation Of Attack Graphs Using NVD," 24th ACM Conf. Comput. Commun. Secur., pp. 135–142, 2018, doi: 10.1145/3176258.3176339.

[19] L. Wang, T. Islam, T. Long, A. Singhal, and S. Jajodia, "An attack graph-based probabilistic security metric," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 5094 LNCS, pp. 283–296, 2008, doi: 10.1007/978-3-540-70567-3_22.

[20] K. Kaynar, "A taxonomy for attack graph generation and usage in network security," J. Inf. Secur. Appl., vol. 29, pp. 27–56, 2016.

[21] K. Kaynar, "Distributed Log Analysis for Scenario-based Detection of Multi-step Attacks and Generation of Near-optimal Defense Recommendations," 2017.

[22] N. C. Idika, "Characterizing and Aggregating Attack Graph-Based Security Metrics," 2010.

[23] S. Y. Enoch, J. B. Hong, M. Ge, and D. S. Kim, "Composite metrics for network security analysis," arXiv Prepr. arXiv2007.03486, 2017.

[24] A. Roy, D. S. Kim, and K. S. Trivedi, "Scalable optimal countermeasure selection using implicit enumeration on attack countermeasure trees," in IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2012), 2012, pp. 1–12.

[25] R. Ortalo, Y. Deswarte, and M. Kaâniche, "Experimenting with quantitative evaluation tools for monitoring operational security," IEEE Trans. Softw. Eng., vol. 25, no. 5, pp. 633–650, 1999, doi: 10.1109/32.815323.

[26] W. Li and R. B. Vaughn, "Cluster security research involving the modeling of network exploitations using exploitation graphs," Sixth IEEE Int. Symp. Clust. Comput. Grid Work. 2006. CCGRID 06, no. July, 2006, doi: 10.1109/ccgrid.2006.1630921.