An Improved Model for Analysis of Host Network Vulnerability

Ramchandra Yadav
PhD. Student, CSE Deptt.
Bundelkhand Institute of Engg.
& Technology, Jhansi, India.

Raghu Nath Verma, PhD
Asst. Professor, CSE Deptt.
Bundelkhand Institute of Engg.
& Technology, Jhansi, India.

Anil Kumar Solanki
Professor, CSE Deptt.
Bundelkhand Institute of Engg.
& Technology, Jhansi, India.

ABSTRACT
With all the news on cyber attacks and computer security in the last few years, it does not take much time to realize that some action must be taken to protect our organization before it hits close to our home. In fact, security has gone from backroom to the boardroom in a lightning speed. Network security depends on most of network configuration and vulnerabilities. Each machines overall susceptibility to attack depends upon the vulnerabilities of another machine. An attacker tries to exploit the least secure system by small attacks iteratively, where each exploit in the network provide the platform for subsequent exploit. Such a series is known as attack path and the set of all possible paths will form an attack graph. By their highly interdependencies, it is much complex to draw traditional vulnerability analysis. Several works have been done to construct an attack graphs. The goal of this paper is to provide a framework, architecture, and an intelligent approach to vulnerability analysis by utilizing the concept of automated scanning tools. By the changing environment, conducting a periodic in-house vulnerability assessment is very much essential.

Keywords
network security, cyber attack, attack graph, vulnerability analysis, vulnerability assessment.

1. INTRODUCTION
There is no method to guess the coming attacks and there is no method to find the time of attacks, we can only reduce the impact of attack by knowing in advance the possible attack. Even well managed networks are also vulnerable to some level of attack, since eliminating all susceptibility to attack is treated to isolating the network, which is not a solution for most of enterprises. Completely dependent on manual processes and mental model is inadequate. There is two automated vulnerability scanning tools Nessus [1] and Retina [2] are available for evaluating path of attacks, for understanding overall security strength.

Attack route are generated by selecting a host and, using network topology information and host vulnerabilities, to conclude how the attacker can take the advantage to compromise vulnerable hosts that are accessible from already compromised hosts. A vulnerability scanner gives few clues as to how attackers might actually exploit vulnerabilities among multiple hosts to advance an attack on a network. In this approach a scanner generate several recommendations to patch the critical vulnerabilities or make firewall configuration more restrictive. In addition, also provides some type of attack graph display. However, the abstract nature of attack graphs, it becomes unmanageable and proven to be a practical weakness in creating an effective display [3].

Network administrators face major difficulties if he faced known software vulnerabilities, which is comes from developer stage. Each year there is hundreds of vulnerabilities discovered in a very short period of time, it is very challenging task for system administrators to monitor and make security precaution for the software running on their host networks. There is only solution to read the bug reports regularly published by open source platforms such as CERT [4], CVE [5], and OSVD [6] etc. that make understanding about the actual security vulnerabilities. Coming each new vulnerability, security assessment is needed to choosing the right countermeasures. Patching, rebooting, reconfigure the firewall rules are major controls to countermeasure vulnerabilities.

2. RELATED WORK
At research level there are several methods have been proposed for analyze the vulnerabilities in network of hosts to construct attack graphs based on data provided by commercial vulnerability scanning tools. Attack paths of potential attacker are identified easily by attack graph. The significant of attack graph analysis [7] is much crucial.

Tito Waluyo Purboyo, Kuspriyanto [8], proposed a model for analyzing the vulnerability. In this context they explain the importance of attack graph, and their analyses rely on accurate model of the network. These models are generally built using raw data from network vulnerability scanners such as Nessus [1].

C. Phillips and L. Swiler [9, 12] proposed a tool for constructing network attack graphs. In our model we use more efficient attack graph representation that makes the graph feasible for larger networks.

S. Templeton and K. Levitt [10] and J. Dawkins, C. Campbell, and J. Hale [11] describe the approaches for identifying and specifying attacks that are similar to our proposed modeling. R. W. Ritchey and P. Ammann [13] proposed an application of model checking very first time.

O. Sheyner, J. Haines, S. Jha, R. Lippmann, J. Wing, [14] describes the modified Symbolic Model Verifier (SMV) model checker to identify the probable attack paths instead of a single attack path. But SMV is a part of TVA analysis engine, though scalability problems arises that need to build a custom analysis engine.

More recently P. Ammann, D. Wijesekera, S. Kaushik [15] was first to describe the application of efficient graph-based representation of exploit dependencies to network vulnerability analysis.

3. TOOLS FOR GENERATING ATTACK GRAPHS

In this Section majorly focus on two important tools as Topological Vulnerability Analysis (TVA) [16] and MULVAL (Multi-host, multistage, Vulnerability Analysis) [17].

3.1 Topological Vulnerability Analysis (TVA)

Modeling tools such as Topological Vulnerability Analysis (TVA) is comes in market to generate attack graphs automatically [18]. Figure 1 shows the overall architecture of TVA tool. There are basically three components: (1) a knowledge base consists modeled exploits. (2) a networks description and (3) a attack target. TVA analysis engine combines these three inputs and then discovers attack paths based on merged model.

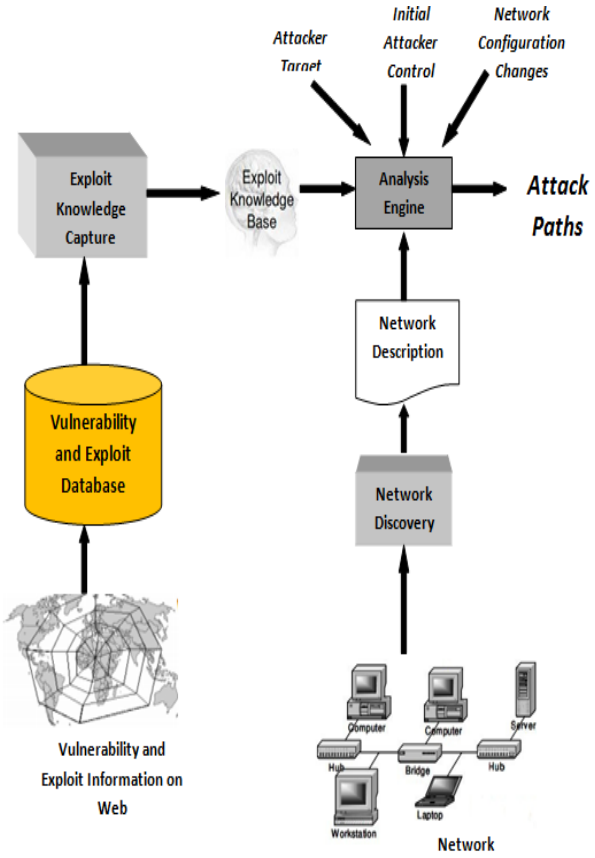

Figure 1: Framework for Topological Vulnerability Analysis (TVA)

Both the network description element and exploit knowledge base have a common name space, which gives the advantages to mapping the common exploits to actual network elements. A network discovery component may have traditional vulnerability scanners, and algorithm to convert tools output to compatible TVA network description. A network discovery component collects system configuration information and connectivity information to gives a TVA network description.

The set of exploits condition in the TVA knowledge base component must be up to date, because discovered attack paths will contains only those exploits that are actually existed in the knowledge base. Once the basic network related information gathered, then we set some rules of exploit pre-condition and post-conditions. These exploit condition are some attributes that potentially impact network security.

3.2 MULVAL (Multi-host, multistage, Vulnerability Analysis)

MULVAL (Multi-host, multistage, Vulnerability Analysis) [17], is a logical programming based tool for network security analyzing. This tool uses the systems configuration information, and logical dependencies between attackers motivation. A logical attack graph directly measure the logical causality relationship among configuration setting and potential attacker privileges. It clearly tells "why an attack can happen", instead of "how an attack happens". The structure of efficient Vulnerability Analysis is shown in Figure 4. The uDrawGraph element gives the graphical output of attack graph. This uDrawGraph is easily available open source software which has multiple functions to view, hide, or zoom in, zoom out graphs or particular part of graphs.

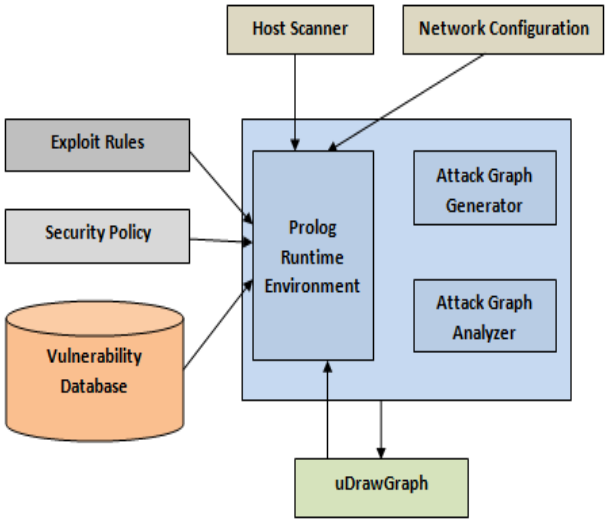

Figure 2: Architecture of Efficient Vulnerability Analysis [8]

Its API provides several customized functionality that will help to easily navigate the attack graph. Additionally users have the option to choose the facts nodes and they can delete or undelete and also observes the impact on the attack graph. After that users can decide whether suitable modification required or not in the actual network configuration [19].

3.3 Intelligent Vulnerability Analysis Model

Attack graph is a graphical representation of all possible attack routes; we see a transition from one state to desired state. The architecture of improved vulnerability analysis model is shown in figure 3.

This architecture contains three modules. The first one is vulnerability scanning module, which scans the all connected host in the network. Second is vulnerability classification module, which categories the discovered vulnerabilities by its set of patterns. Further scanning report divided into two sub-categories as in, application based vulnerability and mis-configuration vulnerability.

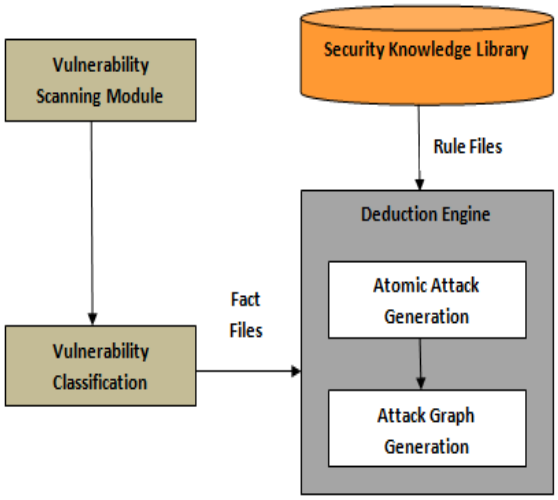

Figure 3: Architecture of intelligent vulnerability analysis model

These set of all raw data are collectively known as fact files and it is forward to deduction engine. The deduction engine module will generate atomic attack and generic attack graphs.

4. ATTACK GRAPH CONSTRUCTION

Attack graph generation was important part of network security, because attack graph visualizes existing vulnerability in short period of time and also gives an idea about how a attacker may exploit the potential vulnerability. For prevention of our enterprise network and implementing suitable security controls we must analyze the attack graph.

Attack graph generation was important part of network security, because attack graph visualizes existing vulnerability in short period of time and also gives an idea about how an attacker may exploit the potential vulnerability. For prevention of our enterprise network and implementing suitable security controls we must analyze the attack graph.

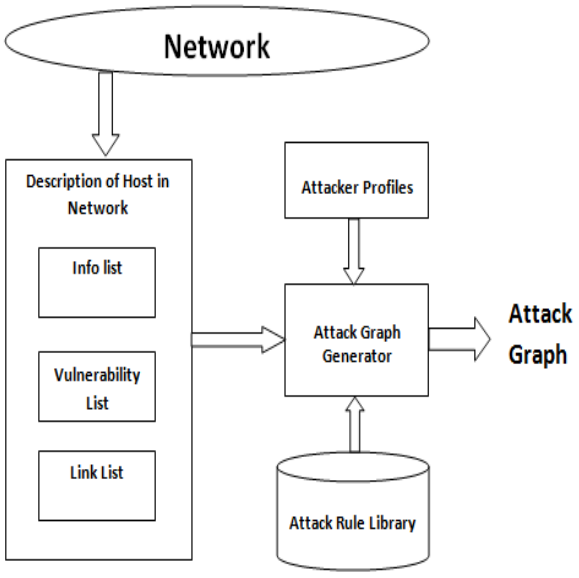

Figure 4: Architecture of Attack Graph generation

Algorithm: AG_Generation (Input)

INPUT: host_attaribute (H), attack _rule (R), initial_status (S0)

OUTPUT: attack graph AG

BEGIN

Step 1. Make the network status queue, named st_que, and add S0 to it.

Step 2. Select next status from st_que. Go to step 3 if this status has not been further carried on, or quit.

Step 3.

a) Take every host as attack source and every host as attack target at a time
b) If the value in Link Matrix for these two hosts (may be is a same host) is 1, then check the Attack Rules condition and identify the eligible attack rules.
c) Execute every attack under these rules and generate a new status at a time. If the new status did not exist in the st_que, then add it to the queue.
d) Generate graphical codes to draw attack edge and nodes from previous status to the new status. The probability of this attack can also been determined from attack rules.
e) Go to step2 after every host visited.

END

Figure 5: Algorithm for attack graph generator process

Fundamental architecture of attack graph generation process is shown in figure 4. We clearly understand that firstly collect information about particular host, it's connected all host, their known vulnerabilities. These all gathered information is forward to attack generator component, which analyze the data with existing attacker profiles and available attack rule library, then visualizes the attack graph. For construction of attack graph, attack graph generator component uses the algorithm described in figure 5.

5. THE PROPOSED FRAMEWORK

With the help of several model studies in this paper, here proposed a new architecture and framework shown in figure 6. Our proposed model will help to analyze the network vulnerability efficiently in a very lesser time. We include the very popular network scanning tool Nessus, which is capable to build the vulnerability data from remote location also. However, the scanning range has a fundamental limitation on the information available about the target host.

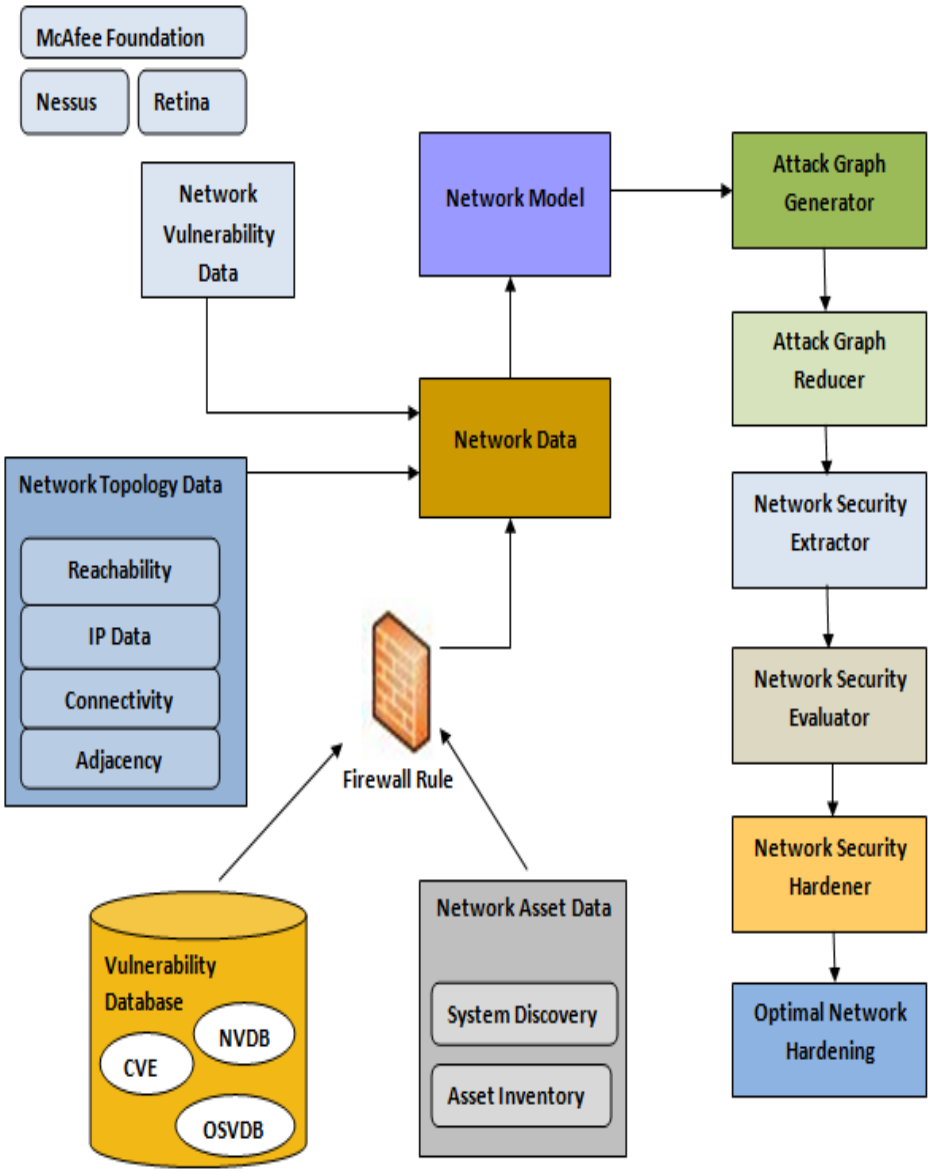

Figure 6: Proposed Architecture for a Network Vulnerability Analysis

6. CONCLUSION

In this paper, we reviewed some of the existing architecture and their perspective for analyzing the vulnerability of computer network. We seen attack graph is important aspect to assess the system security and enterprise policies. With the dynamic nature of the attacks scenario and topological structure changes day by day that will be present major challenges. Some other aspects have been also suggested as to represent the hierarchical attack for network security analysis [20]. Another enhanced method based on aggregating attack graph security metrics was also suggested as in N.C. Idika [21]. The ultimate goal of this paper is to help the system administrator- by giving him a fast and efficient model to test out different system configurations such as network connectivity, firewall rules, services running on hosts and finding new attacks by which system is vulnerable. This paper will help for researcher to understand the concept of attack graph modeling, its construction and making policies to defend the cyber attacks.

7. REFERENCES

[1] Nessus Open source vulnerability scanner project. http://en.wikipedia.org/wiki/Nessus_(software), 10-07-2016.

[2] Retina Security Scanner. https://www.beyondtrust.com/products/retina-network-security-scanner, 06-07-2016.

[3] L. Williams, R. Lippmann, K. Imgols, "An Interactive Attack Graph Cascade and Reachability Display," VIZSEC 2007.

[4] http://www.cert.org/vulnerability-analysis/knowledgebase/index.cfm, 15-07-2016.

[5] Common Vulnerabilities and Exposures (CVE), https://cve.mitre.org, 15-07-2016.

[6] Open Source Vulnerability Database(OSVD), http://osvdb.org

[7] L. Wang, S. Jajodia, A. Singhal, P. Cheng, and S. Noel, "K-zero day safety: a network security metric for

mearsuring the risk of unknown vulnerabilities," IEEE Transaction on Dependable and Secure Computing, vol. 11, no. 1, pp. 33-44, 2014.

[8] Tito Waluyo Purboyo, Kuspriyanto, "A Framework for Analysis A Network Vulnerability," IJETTCS, vol. 2, pp. 405-409, August 2013.

[9] C. Phillips and L. Swiler, "A graph-based system for network vulnerability analysis," In Proceedings of the New Security Paradigms Workshop, pp. 71–79, Charlottesville, VA, 1998.

[10] S. Templeton and K. Levitt, "A requires/provides model for computer attacks," In Procesdings of the New Security Paradigms Workshop, cork, Ireland, September 2000.

[11] J. Dawkins, C. Campbell, and J. Hale, "Modeling network attacks: Extending the attack tree paradigm," In workshop on Statistical and Machine Learning Techniques in Computer Intrusion Detection, Johns Hopkins University, June 2002.

[12] L. Swiler, C. Phillips, D. Ellis, and S. Chakerian, "Computer-attack graph generation tool, "In Proceedings DISCEX '01: DARPA Information Survivability Conference & Exposition II, pp 307–321, June 2001.

[13] R. W. Ritchey and P. Ammann, "Using model checking to analyze network vulnerabilities," In Proceeding of the 2000 IEEE Symposium on Security and Privacy (Oakland 2000), pp 156-165, Oakland, CA, May 2000.

[14] O. Sheyner, J. Haines, S. Jha, R. Lippmann, J. Wing, "Automated Generation and Analysis of Attack Graphs," in Proceedings of IEEE Symposium on Security and Privacy, Oakland, California, May 2002.

[15] P. Ammann, D. Wijesekera, S. Kaushik, "Scalable, Graph-Based Network Vulnerabity Analysis," in Proceedings of CCS 2002: 9th ACM Conference on Computer and Communications Security, Washington, DC, November 2002.

[16] S. Noel, M. Elder, S. Jajodia, P. Kalapa, S. O'Hare, K. Prole, "Advances in Topological Vulnerability Analysis," IEEE CATCH 2009.

[17] X. Ou, S. Govindavajhala, A.W. Appel, " MulVAL: A Logic-based Network Security Analyzer," In SSYM'05: Proceedings of the 14th conference on USENIX Security Symposium, pp 8-8, Berkeley, CA, USA, 2005.

[18] S. Jajodia, S. Noel, "Topological Vulnerability Analysis : A Powerful New Approach for Network Attack Prevention, Detection, and Response," Indian Statistical Institute Monograph Series, World Scientific Press, 2008.

[19] D. Saha, "Extending Logical Attack Graphs for Efficient Vulnerability Analysis," in Proceedings of CCS'08: 15th ACM conference on Computer and Communications Security, Alexandria, Virginia, USA, October 27-31, 2008.

[20] J. Hong and D.S. Kim, "HARMs: Hierarchical Attack Representaion Models for Network Security Analysis," SRI Security Research Institute, Edith Cowan University, Perth, Australia, 2012.

[21] N. C. Idika, "Characterizing and aggregating attack graph-based secuurity metric [Ph.D. thesis]," Center for Education and Research, Information Assurance and Security, Purdue University, 2010.