Volume 4, Issue 6, June 2014 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Securing Network with Hicuts, Attack Graphs and Quantification using Security Metrics

Swarupa V. Mohite
M.E. Final Year student,
Department of Computer Science and Engineering,
D. Y. Patil College of Engineering and Technology.
Kolhapur, India

G. A. Patil
Department of Computer Science and Engineering,
D. Y. Patil College of,
Engineering and Technology
Kolhapur, India

Abstract— Nowadays as the security threats in network are increasing, multistage and multiple host attack scenarios must be considered while assessing the network vulnerability towards attacks. One of the way to network vulnerability problem is to construct an attack graph for network configuration. Attack graph consists of number of attack paths which are series of exploits. And with the help of these paths, attacker tries to reach the destination. Each attack path represents an attack scenario. As the number of attack scenarios increase, the overall security of the network reduces. Thus there is need for quantification of security level of a given network. In this paper, a security approach is provided and two security metrics, namely probabilistic security metric and attack resistance metric, have been used to evaluate the relative security levels of network configurations. A case study has been presented to demonstrate the applicability of the proposed approach.

Keywords— Packet Classification, HiCuts, Attack Graph, Probabilistic Security Metric, Attack Resistance Metric

I. INTRODUCTION

Network is a system of interconnected computers. There are large networks and small networks, but size of network is irrelevant in terms of importance of network security. The intention behind network security is to protect the network and its component parts from unauthorized access and misuse. Network security system is an essential component of the configuration as well as network management. Implementation of effective network security provides both physical and information security to paths, links and databases. Network security is a technique which protects the basic networking infrastructure from an unlawful access, malfunction, wrong usage, damage or alteration. This gives a protected platform for computers, programs and various users to execute their permissible vital functions within a secured environment.

Present day network security scanners are available which are able to detect vulnerabilities local to a system. But these scanners have limitations that, they are not efficient to identify multi-stage and multi-host attacks[1]. In practice, some vulnerability may still remain in a network after they are discovered, due to environmental factors and cost factors. To remove such remaining vulnerabilities there is a need to assess and quantify the possibility that attackers may compromise significant resources through combining various vulnerabilities.

In network security the current focus is on the qualitative nature of security, rather than quantitative study of network security. Assessing the overall security of a network requires a thorough understanding of the interplay between host vulnerabilities. Such an understanding is difficult to obtain with vulnerability scanners and intrusion detection systems. These tools focus on identifying individual vulnerabilities or attacks, and are usually unaware of the relationships among vulnerabilities or attacks.

One such tool that gives description about the correlated attacks is attack graph[1][3]. An attack graph consists of a number of attack paths each of which depicts an attack scenario. Therefore, more the number of attack paths and attack scenarios, higher is the probability of compromising a target. Thus, there is need for quantification of security level of a given network.

In the proposed system work, packet behavior and characteristics are studied, analyzed and used for detection of suspicious and unknown packets. Network packets are captured and examined. To identify the packets, packet classification algorithm HiCuts is implemented. Attack graph is constructed to show attack paths with more clear representation. The result of implemented work is showing detected suspicious packets and safe packets, action taken on suspicious packets and security strength of network. In next section II we are presenting the literature survey over various methods of network configuration. In section III, the proposed approach and its system architecture diagram is depicted. In section IV we are presenting the current state of implementation and results achieved. Finally conclusion and future work is presented in section V.

II. LITERATURE SURVEY

- According to Nirnay Ghosh., S. K. Ghosh [1], many network security scanners are available and are efficient as far as detecting vulnerabilities local to a system is concerned. But they do not identify all conditions for a complete attack to take place, or how different vulnerabilities existing on different systems may be combined to produce multi-stage, multi-host attacks.

- The work has been done by Balzarotti et al [2] where they were using functions for modeling the effect of executed exploits on the resistance value of other exploits. However, their work focuses on computing the minimum effort required for executing each exploit, whereas it needs to compute the overall security of a network with respect to given critical resources.
- Assessing the overall security of a network requires a thorough understanding of the interplay between host vulnerabilities. That is, which and how vulnerabilities can be combined for an attack. L. Wang, A. Singhal, and S. Jajodia [3] concludes that the existing tools typically focus on identifying individual vulnerabilities or attacks, and are usually unaware of the relationships among vulnerabilities or attacks so there is need to detect such correlated attacks.

III. PROPOSED APPROACH

Proposed security system works for training and testing system for dataset, generating attack graphs, which will detect, correct and prevent existing network from attacks and analyse the security status of a network. Figure 1 shows Architecture of proposed security system. It has five steps as identifying individual exploit conditions, identifying relations between exploits of different nodes, representing attack graphs, performing respective preventing, corrective actions for the detected attacks and quantifying the security strength of network.

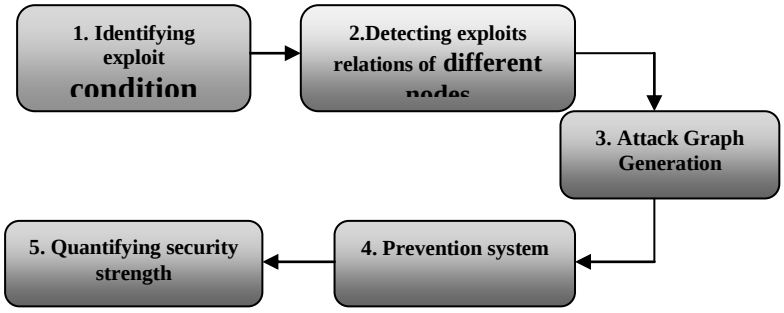


Fig 1.Architecture of proposed security system

It is a mechanism which is present on each host in the network, and will find the exploit conditions based on conjunctive and disjunctive relations between vulnerabilities and generate its attack graph. This attack graph will show correlations of exploits of various nodes and will help in performing preventive and corrective measures for those exploit conditions. Total security level of network configuration is analysed using various security metrics.

IV. SYSTEM DESIGN

A. *Identifying exploit condition:-*

For identifying exploit conditions, packet capturing, training and testing activities are done. Firstly each node in the network is trained with safe data transactions as training activity. From the training data rules are created as

@192.168.1.3/32 192.168.1.1/32 375 : 383 375 : 383 0x06/0xFFFF

@192.168.3.141/32 192.168.3.141/32 0 : 100 0 : 100 0xFF/0xFF

These rules indicates source IP address, destination IP address, Packet size at source, Packet size at destination, ports used etc. These rules are stored and used further in detection process. Thereafter, system creates HiCuts decision tree internally using packet classification algorithm. All the rules are stored with this tree and then onwards this tree works as a model for testing phase i.e. exploit condition detection mode. Each time a packet arrives in network, already created decision tree is traversed to find a leaf node. Each leaf node stores a small number of rules. Linear searches among these rules yield the packet parameters matching. Packets with matched rules are known to be safe and hence sent to destination and others are suspicious and hence blocked and represented with red colour. Likewise the system is enabled to identify individual exploit conditions on single node.

B. *Detecting exploit relations of different nodes:-*

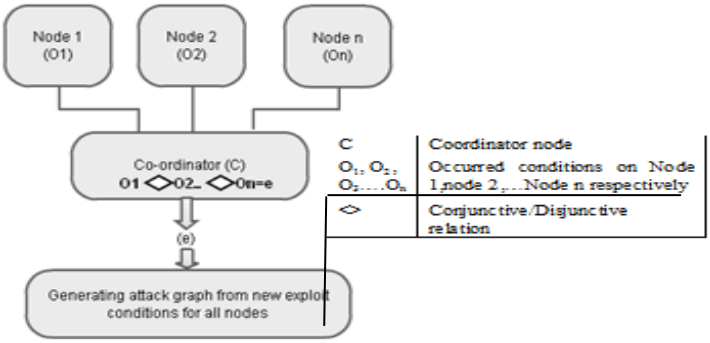


Fig. 2 Generation of new exploit conditions by combination of existing conditions.

Figure 2 shows the process of selecting one node as a coordinator node among all the existing nodes. The role of a coordinator is to identify the combinations of conditions occurred on multiple nodes which will generate some new exploits and are called multi-host attacks.

Every node has its own list of occurred exploits, which the nodes can send to attack graph generation module. For detection of combined exploit conditions, coordinator must update with the exploit lists of all other nodes. So for this purpose each node sends its current exploit list to coordinator. The coordinator will check for possible relations between all combinations of exploits and conditions. The relation can be conjunctive or disjunctive relation. The Coordinator synchronizes all exploit lists. And then new updated exploit data is sent to all the client nodes to improve with their further detection process.

C. ***Attack Graph Generation with attack Prevention and correction process:-***

According to the process of detection of individual exploit and relations of exploits, respective actions are taken on safe and unknown packets. All the matched packets are allowed inside and suspicious unknown packets are blocked in the system. Blocked packets are represented with red color and safe are shown with white color.

On the basis of this data, complete attack graph is generated. An attack graph is actually achieved by connecting all individual nodes' graph. The generated attack graph will be used to identify the vulnerabilities leading to possible attacks and possible cumulative attacks. It is also used to identify the measures for further attacks [4].

From the attack graph , we come to know that which nodes are affected by exploits, which nodes are source of the exploits or through which intermediate nodes exploit conditions are propagated to target node.

D. ***Testing network security and performance:-***

The implemented system is tested with the conjunctive and disjunctive relations between the exploit conditions. Analysis of obtained results elaborates the effect of system on network security and its performance. The attempt to mathematically quantify the security strength of a network system is based on security metrics like probabilistic security metric and attack resistance metric.

Wang et. al. [3] have proposed a metric called as probabilistic security metric; which quantifies the probability of successfully executing an exploit and measures the likelihood of compromising a network in terms of the fraction of attackers reaching the goal. Therefore, this metric can be used to measure the degree of security strength for a network configuration. For an exploit e and condition c, the probabilistic security metric is given by two scores [6]-

• Individual score- It defines the intrinsic likelihood of execution of an exploit or a security condition, denoted by p(e) and p(c) respectively. This score is assigned based on expert knowledge about the vulnerability.

• Cumulative score- It measures the fraction of attackers who successfully reach an exploit e or a condition c, denoted by P(e) and P(c) respectively. This score is evaluated using the probabilities of independent and non-mutually exclusive events. For an exploit *e* if preconditions *c*1 and *c*2 are required to be satisfied simultaneously, the cumulative score for the exploit e is given by

$$P(e) = P(c1).P(c2).p(e)$$

Similarly, if a post condition c requires either of the exploits e1 and e2 or both to be satisfied, the cumulative score for the condition c will be given by the formula for calculating the probability of occurrence of two non-mutually exclusive events as,

$$P(c) = p(c).\ (P(e1) + P(e2) - P(e1).P(e2))$$

An attack resistance metric [3] of a network configuration is a composition of measures of individual exploits. The resistance of an exploit is interpreted as the effort that an attacker requires to put in until success i.e. successful execution of the exploit. Reciprocal of probability of success gives the number-of-attempts (effort) required until success is achieved. Two basic operators for computing *attack resistance* of an exploit are as follows:-

$\oplus$ operator - This operator is used to realize the scenarios where a disjunctive relationship exists between two or more exploits to successfully execute a different exploit. If r1 and r2 are the individual attack resistances of two exploits e1 and e2 respectively then,

$$r1 \oplus r2 = \frac{1}{\frac{1}{r1}+\frac{1}{r2}}$$

$\otimes$ operator - It is used to realize the scenarios where a conjunctive relationship exists between one or more exploits for successful execution of another exploit. If r1 and r2 are the individual attack resistances of two exploits e1 and e2 respectively then,

$$r1 \otimes r2 = r1 + r2$$

R- Cumulative attack resistance (R) values are calculated using individual attack resistances r which are generated after executing each instantiated exploit. Likewise using these security metrics, P(C) and R values are calculated, which represents final security strength of the current network.

V. IMPLEMENTATION

Using WinPcap all the packets of safe transactions are captured and proposed system is trained with the safe data. After training, duplicates are removed and data is normalized. By analyzing packet parameters as ranges of port values used, packets sizes, protocols used, source IP's, destination IP's; rules are created. Packet classification is done using Hierarchical Intelligent cuttings packet classification algorithm which is a multi-dimensional packet classification algorithm[7]. It preprocesses the packet classification rules to build a decision-tree for field-dependent search, and in each leaf-node of the decision-tree, a small number of rules bounded by a threshold. After training, system works in detection mode. Every time a packet arrives in network, already created decision tree is traversed to find a matching leaf node. Linear searches among these rules yield the packet parameters matching. System is trained such that the packets with known and acceptable parameters are only delivered to desired destination and rest will be blocked.

For detection of combined exploit conditions, coordinator gets updated with new rules i.e. system gets re-trained with the exploit lists of all other nodes automatically after every 10 seconds. The coordinator will check for possible relations between all combinations of exploits and conditions and will detect new generated exploits. This new generated exploit data is sent to all the clients to improve their detection process.

The results of exploit detection process are sent to Attack graph generation [9]. Generated attack graph is used to identify the vulnerabilities leading to possible attacks and possible cumulative attacks. And it is also helpful for identifying measures for further attacks.Finally mathematical quantification of security strength of a network is done by calculating security metrics like probabilistic security metric[5] and attack resistance metric.

VI. RESULTS OF PRACTICAL WORK

Following figures show Results of Practical Work

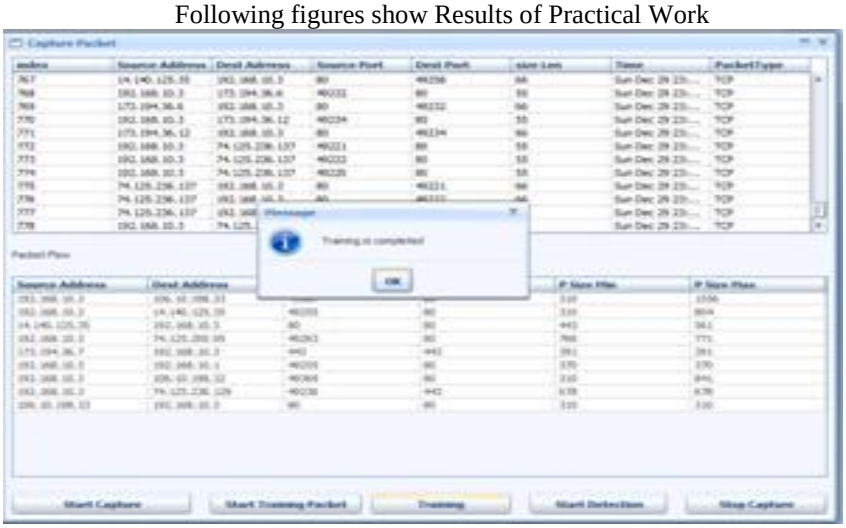
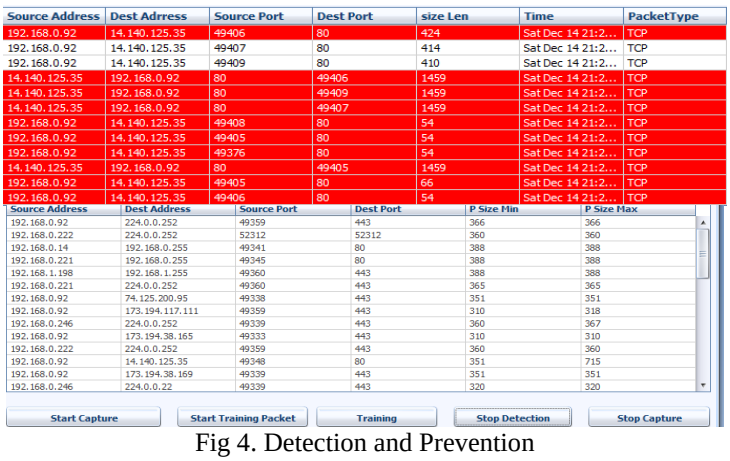


Fig 3. Central training

Individual nodes in the network are trained and tested for vulnerabilities and then for detection of combined exploit conditions, coordinator gets update with the exploit lists of all other nodes after every 10 secs. This new generated exploit data is sent to all the clients to improve their detection process i.e. all clients synchronizing rules with coordinator. This central training process is shown in Figure 3.



Source Address	Dest Adrress	Source Port	Dest Port	size Len	Time	PacketType
192.168.0.92	14.140.125.35	49406	80	424	Sat Dec 14 21:2...	TCP
192.168.0.92	14.140.125.35	49407	80	414	Sat Dec 14 21:2...	TCP
192.168.0.92	14.140.125.35	49409	80	410	Sat Dec 14 21:2...	TCP
14.140.125.35	192.168.0.92	80	49406	1459	Sat Dec 14 21:2...	TCP
14.140.125.35	192.168.0.92	80	49409	1459	Sat Dec 14 21:2...	TCP
14.140.125.35	192.168.0.92	80	49407	1459	Sat Dec 14 21:2...	TCP
192.168.0.92	14.140.125.35	49408	80	54	Sat Dec 14 21:2...	TCP
192.168.0.92	14.140.125.35	49405	80	54	Sat Dec 14 21:2...	TCP
192.168.0.92	14.140.125.35	49376	80	54	Sat Dec 14 21:2...	TCP
14.140.125.35	192.168.0.92	80	49405	1459	Sat Dec 14 21:2...	TCP
192.168.0.92	14.140.125.35	49405	80	66	Sat Dec 14 21:2...	TCP
192.168.0.92	14.140.125.35	49406	80	54	Sat Dec 14 21:2...	TCP

Source Address	Dest Address	Source Port	Dest Port	P Size Min	P Size Max
192.168.0.92	224.0.0.252	49359	443	366	366
192.168.0.222	224.0.0.252	52312	52312	360	360
192.168.0.14	192.168.0.255	49341	80	388	388
192.168.0.221	192.168.0.255	49345	80	388	388
192.168.1.198	192.168.1.255	49360	443	388	388
192.168.0.221	224.0.0.252	49360	443	365	365
192.168.0.92	74.125.200.95	49338	443	351	351
192.168.0.92	173.194.117.111	49359	443	310	318
192.168.0.246	224.0.0.252	49339	443	360	367
192.168.0.92	173.194.38.165	49333	443	310	310
192.168.0.222	224.0.0.252	49359	443	360	360
192.168.0.92	14.140.125.35	49348	80	351	715
192.168.0.92	173.194.38.169	49339	443	351	351
192.168.0.246	224.0.0.22	49339	443	320	320

Start Capture | Start Training Packet | Training | Stop Detection | Stop Capture

Fig 4. Detection and Prevention

As shown in Figure 4 system works in detection mode. Every incoming packet is compared with generated rules after training, by traversing the HiCuts decision tree and only packets with matching rules are safe so delivered to appropriate destination and shown in white, while others are suspicious so blocked and marked with Red color.

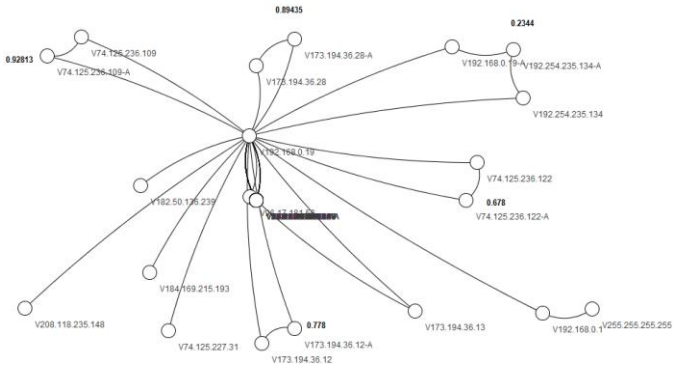


Fig 5. Generated attack graph

Figure 5 shows attack graph for current state scenario with both safe and suspicious communication. Here, vertices of attack graph are divided into three categories:-server, clients and address of website communicated. To show the safe communication, edge connects vertices indicating client and website address. And to show suspicious communication or attacking condition, edge connects vertices represented with client and website address which is appended with –A .

If any communication has some of the safe transactions and some of the attacking transactions then vertices are duplicated and one is showing just the address of communicated website and other appended with –A.

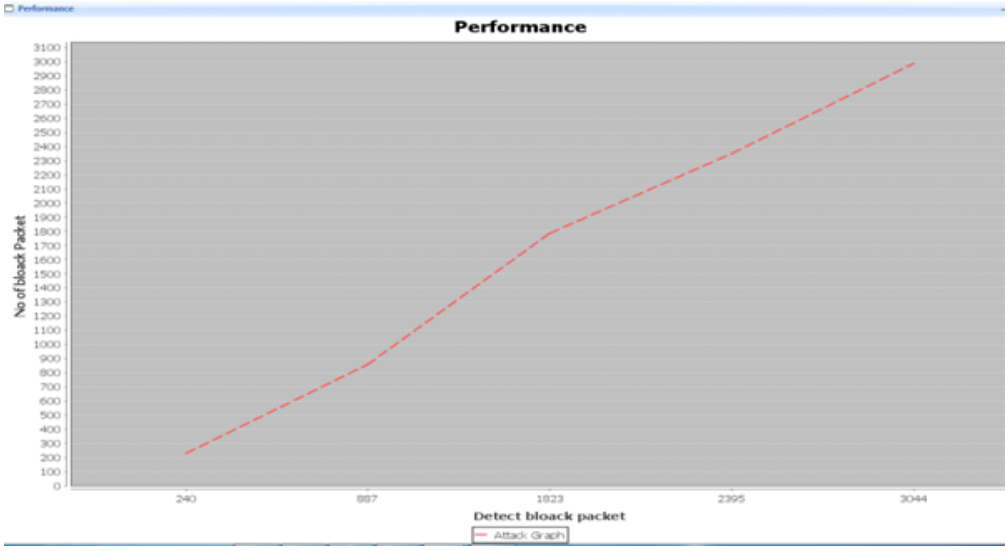


Fig 6. Performance graph

The number of suspicious packets detected Vs actually blocked packets are represented using performance graph as shown in Figure 6.

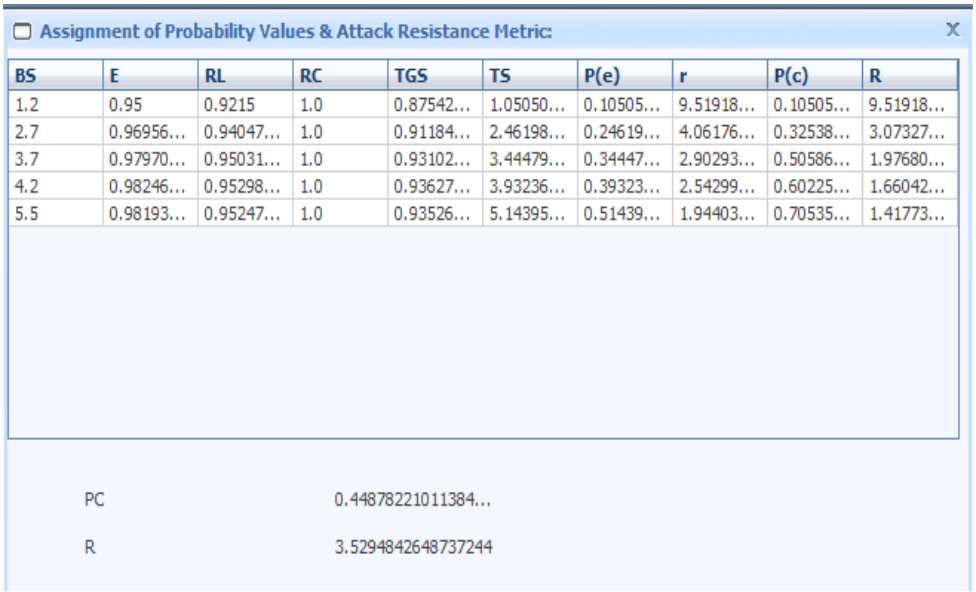


Fig 7.Quantifying the security strength with P(C)and R

By calculating probabilistic security metric and attack resistance metric average P(C) and R values are derived as shown in Figure 7. The cumulative probability score P(c) is calculated for all the nodes of the attack graph to determine the fraction of attackers successfully compromising the goal and cumulative resistance (R) for each attack path reaching the goal can be computed by simply adding individual resistance values along the path. So the end values are :

P(C) = 0.448

R = 3.52

Result shows that cumulative resistance of the whole network should be smaller than the cumulative resistance of each possible attack path. If number of attack scenarios is less then it offers more resistance to external attacks.

VII. CONCLUSION AND FUTURE WORK

This system is using the technique in which the continuous monitoring of incoming packets is done. Systems are trained and tested for HiCuts. It detects attacks by considering relations between the exploits on different nodes and generates attack graph. Security is quantified as we can get the values for probability of successfully executing an exploit and fraction of attackers reaching the goal. So result of the system is secured network for individual and related attacks.

In future there is a room to improve the performance of the system by improving the technique and algorithm we applied. Like HiCuts packet classification algorithm can be improved for memory usage, classification speed and lower run-time. Attack graphs also has scope to improve because it has disadvantage regarding complexity in visualization. As the number of hosts and vulnerabilities increases, the complexity of attack graphs boosts rapidly, preventing the administrator from understanding the graph and extracting remedies manually. So attack graphs can be replaced with Attack grammar [10]. Our aim is to improve the performance of the system and to add more features to find novel attacks.

REFERENCES

[1] Nirnay Ghosh., S. K. Ghosh . :An Approach for Security Assessment of Network Configurations using Attack Graph: First International Conference on Networks & Communications(2009).

[2] Balzarotti, D., Monga, M., Sicari S.: Assessing the risk of using vulnerable components. In: Proceedings of the 1st Workshop on Quality of Protection (2005)

[3] L. Wang, A. Singhal, and S. Jajodia. : Measuring the overall security of network configurations using attack graphs. In Proceedings of the 21st Annual IFIP WG 11.3 Working Conference on Data and Applications Security (DBSec), July 2007

[4] Nirnay Ghosh, S. K. Ghosh. :An Intelligent Technique for Generating Minimal Attack Graph. In proceedings of the 21st annual computer security applications conference(ACSAC 2005)

[5] L. Wang, T. Islam, T. Long, A. Singhal, and S. Jajodia. : An attack graph-based probabilistic security metric. In Proceedings of the International Federation for Information Processing (IFIP), 2008.

[6] M. Schiffman. Common vulnerability scoring system (cvss).

[7] Pankaj Gupta and Nick McKeown. :Packet Classification using Hierarchical Intelligent Cuttings

[8] Yaxuan Qi, Jun Li. : Performance Evaluation and Implementation of Algorithmic Approaches for Packet Classification

[9] Tito Waluyo Purboyo and Kuspriyanto. : Some Algorithms for Generating Attack Graph. In International Journal of Advanced Research in Computer Science and Software Engineering.Volume 2, Issue 8, August 2012

[10] Yinqian Zhang, Xun Fan, Yijun Wang, Zhi Xue. : Attack Grammer-A New Approach to Modeling and Analyzing Network Attack Sequences. In 2008 Annual Computer Security Applications Conference

