A Study of Network Security Metrics and Secure Computing

Swetha.S[1], Kalpana.B[2], Janani.D[3] , Divya Devi.P[4]

[1]*IVYear, Student, Department of CSE, Panimalar Institute of Technology, Tamilnadu, India.*
[2]*Assistant Professor ,Department of CSE, Panimalar Institute of Technology, Tamilnadu, India.*
[3]*IVYear, Student, Department of CSE, Panimalar Institute of Technology, Tamilnadu, India.*
[4]*IVYear, Student, Department of CSE, Panimalar Institute of Technology, Tamilnadu, India.*

ABSTRACT

Dependable and secure computing has grabbed the spotlight and has pioneered the fusion between dependability and security since 2000. The field of dependability has progressed to be an internationally active field of research. The objective is to explicate a set of attributes that includes reliability, security, safety, availability, confidentiality, integrity and maintainability. Due to the increasing reliance on computer systems in most societies, the field is of growing importance. It has control over the physical access to the hardware and also provides protection against harm that may come via network access and code injection. Malpractice done by operators, whether intentional, accidental, or due to them being tricked can also be controlled. Also the rapid increase in mobile applications, computers, and wireless networks has modified the characteristics of network security. A series of Internet attack and fraudulent acts on individual network and companies have shown that open computer networks have no immunity from intrusions. The purpose of the paper is to provide review of the existing literature available in the dependable and secure computing domain. The paper also presents an overview and study of the network security metrics with several security threats, security issues, security solutions and currently used methodologies to overcome it.

Keywords: - *Dependability, Vulnerability, Security, Threats*

1. INTRODUCTION

As interest in system reliability and fault tolerance increased in the 1960s and 1970s, dependability came into existence encompassing additional measures like safety and integrity. Dependability and security has a major importance in both government and commercial sectors. Dependability also increases interest in verification, validation and measurement of various aspects of system survivability, performance, fault tolerance and safety [1]. With the advent of Internet, Computer networks have grown both in complexity and size. It facilitates easy access to vast store of collaborative computing, reference materials and information sharing. Security mechanism is enforced by authentication and access control. However, these security practices do not take network service-based or application vulnerabilities and operating systems into account.

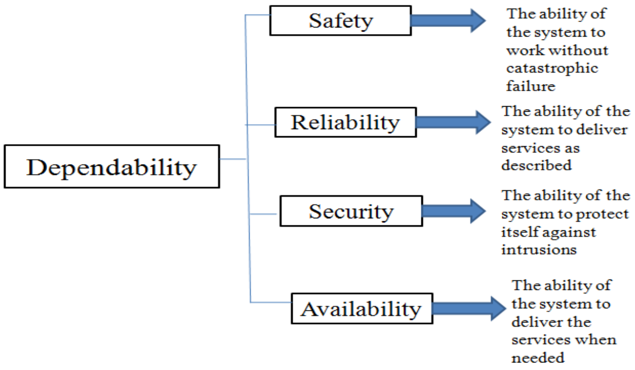

Dependability
Safety
The ability of the system to work without catastrophic failure
Reliability
The ability of the system to deliver services as described
Security
The ability of the system to protect itself against intrusions
Availability
The ability of the system to deliver the services when needed

Figure 1: Dependability and Security Tree

Attackers exploit these vulnerabilities bypassing the authentication policies and access control and can gain legitimate access to resources in the network with the usage of hacking tools. For today’s increasingly complex network environment, the traditional binary view of network security is becoming less suitable. Attack graph presents a representation of various attack scenarios that is specific to a network. Attack graph depicts all available multi-step attack scenarios which an attacker can create and models service or application based attacks. An Attack Path describes an attack scenario that results in computing organization values. It tells us how an attacker gets access to the victim computer; which and how vulnerability attacker can take advantage of and what sort of damage may be done that will impact the organization. Some vulnerability may not seem significant, when defending an isolated network with resources that are critical. By exploiting related vulnerabilities sequences, attackers can often intrude a well-guarded network using multi-step attacks. Attack graphs can identify such potential threats by calculating all possible sequences attackers. To protect critical resources, it is necessary to identify the probability of multi-step attacks in today’s network environment. Attack graphs also specify the missing information about network components and thus allow us to take into account the potential attacks and its further consequences. Such methodology makes it possible to compose individual measures of resources, vulnerabilities, and configurations into a total measure of network security. By computing the risk for enterprise networks, attack graphs allow the administrator to understand the dangerous threats and select the most effective solutions. It is not easy for a human to determine which configuration settings should be modified to describe the identified security problems. It is hard for a human user to measure possible configuration changes and to verify changes made without clear observation of the existing system. The attack graph generation requires knowledge about the effect of exploitation on the network and attacker privileges. However, while attack graphs finds out the threats, but they do not provide a solution to network hardening them. The severity associated with each attack scenario can be measured through some attack graph-based security metrics.

Network security metrics are also essential to enable quantification of the degree of freedom from possibility of undergoing loss or damage from malicious attacks. Security metrics are needed to compare the effectiveness of different security solutions. A zero-day attack is a cyber attack that exploits vulnerability and has not been disclosed publicly. There is no defence against a zero-day attack. It is known as a "zero-day" because once the flaw becomes known; the admin has zero days in which to plan against its exploitation. The vulnerability remains unknown, the fault cannot be patched and anti-virus products cannot identify the when all users of software install the patch to fix the vulnerability. Reliability plays a vital role and proper functioning of a large number of interconnected attacks through signature-based scanning. The first phase of vulnerability begins when it is identified by the third-party software analyst, vendor or the hacker. The security risk associated with vulnerability is particularly high if it is first discovered by hackers. The next phase starts with the public disclosure of the vulnerability, which is repeated by a hacker, any third-party software analyst or a hacker. The information about vulnerability is freely available to everyone after disclosure; therefore, the security risk level increases further because the hacker community is active in zero-day exploits release. The aim is to provide a solution for the vulnerability as soon as possible. The life cycle of a vulnerability ends information systems is also needed. Since “you cannot improve what you cannot measure”, it is important to measure the amount of security provided by different configurations in order to improve the security of these systems [2].

2. OVERVIEW

2.1 Attack Graph-Based Security Metrics

Patil *et al.* [3] suggests a way to network attack modelling and to evaluate security. The paper is based on modelling of computer networks, building attack graphs, processing alert mechanisms for real-time adjusting of specific attack graphs, evaluating various security metrics. These systems also carry with them the dangers of insecurity. Most of the existing metrics have lacked a significant experimental validation and a sound theoretical basis. In spite of these problems, the judicious methodical application of software metrics can significantly improve software quality and productivity. Providing security to the software systems serves as one of the most prominent challenges confronted by people today and hence it is worth exploring the metric analysis to assist the software engineers. The attack based graph is used to render security to the network. The security metrics that can obtain security-relevant information are the number of paths metric, the shortest path metric, and the mean of path lengths metric. The KCA metric may degenerate to the Shortest Path metric when a goal state exists and the semantics of the attack graph are such that the goal state contains all of the values in the network. Consider if the attacker can achieve the goal state in one step and the non-attacker nodes has lesser importance when compared to the target node, then using the Shortest Path metric without the KCA metric would be enough to determine which of the two networks has more security. The mean of path lengths metric and the shortest path metric fail in the number of ways and the security policy may be violated by the attacker. The Number of Paths metric fails to consider the attack effort with respect to the attack paths. To overcome these limitations, an attack graph-based security metrics is proposed and an algorithm is described for merging the operations of these metrics. Attack graph can also provide hints for the network defender on how the vulnerability on the network is exploited by the attacker in order to achieve the goals. Providing Network Security is a very challenging task in large organizations. Attack graphs play a vital role in securing the network because it can find out the presence of vulnerabilities in network and how attackers make use of the vulnerabilities to execute an effective attack. The paper specifies efficient algorithms that can be used to generate the attack graph.

A set of goal conditions and initial conditions are required to generate an attack graph. Network states are the initial conditions and it is available by default. The various directions in evaluating network security are simulation of available malefactor's actions, the attack graph representation of these actions, verification of various properties of the graphs, and determination of security metrics which can explore possible ways to increase security level.
Metrics are developed based on the points of view as specified in the following explanation. A metric is a standardised way of measurement. A good metric should be
a) Consistently evaluated, without subjective criteria.
b) Very cheap to gather, preferably in an automated way.
c) Viewed as a percentage or number, not with qualitative labels like "high," "low" or "medium."
d) Specified using at least one unit of measure, such as "hours," "dollars".
e) A good metric should also ideally specific so that decision-makers can take action.

The advantages of using Attack Graph based security metrics are computing different security metrics and obtaining security assessment procedures. One tool to work with various attack scenarios specific to a network is attack graph. Security to the computers from unwanted threads can be provided by three path-analysis attack graph-based security metrics. By increasing the count, security metrics that provide unique security-relevant information will enhance the security engineer's ability to assess a Network's security and to perform network hardening. However, the metrics doesn't include the developing enhanced approaches for quantitatively measuring attack path complexity. Allowing the people to make use of software systems also serves as a major challenge for software engineers.

2.2 Network Security Metrics- A Review

Tito Waluyo Purboyo, Kuspriyanto [4] proposes a way to evaluate an enterprise network accurately, a graph based on an attack is effectively used by a network administrator. The critical threats are measured in depth and the best countermeasures are chosen by the network administrator. Attack graphs are used to gain knowledge about vulnerability of the systems and to choose what security measures are needed to deploy them. An abstraction that represents how an attacker may ignore security policy by exploiting the interdependence between the known vulnerabilities is referred to as an attack graph. Attack graph-based security metrics is the analyses of attack graph whose job is to obtain data related to security from the attack graph. Attack graphs are a valuable tool to network administrators because it describes the paths which can be used by an attacker to obtain access to a targeted network. Network administrator can then focus their efforts on configuration errors and correcting the vulnerabilities. Attack graphs are used to find out if required goal states can be reached by attackers trying to enter into the enterprise networks from initial states. The starting node denotes an attacker at a given network location. Nodes and arcs specify actions the attacker takes and modifications in the network state caused by these actions. An action usually consists of exploits or exploits steps which take the benefit of vulnerabilities in the software. The attacker has to obtain the restricted privileges on target hosts. The target could be a router, a computer user or a firewall.

Actions are used as stepping stones to reach the target host in large attack graphs. A complete attack graph describes all possible sequences of attacker actions that finally results in the required level of the target privilege. Nodes are used to represent network states and arcs are used to represent attack actions. Some researchers use other representations in which arcs are used to represent network states and nodes are used to represent actions. Attack graphs can have one source and target host or multiple source and target hosts. Several kinds of analysis on configuration-specific attack graphs can be performed by the system administrator to satisfy the security needs of the network. The K-step Condition Accumulation (KCA) metric specifies the ―effectiveness" the attacker can achieve on a network in K steps. Effectiveness represents the capability an attacker attains. Access controls are used to control the capabilities. Therefore, effectiveness can be represented by the privilege(s) the attacker achieves on a machine. Hence, if an attacker can achieve more capabilities on System2 than the attacker can on System1 in K steps, then System1 provides more security than System2. If an attacker can obtain strictly more effectiveness in System2 than in System1 in K steps, then System1's security is higher than System2. Experience has shown that organizations are unwilling to change their security policy even in the face of evidence that their network is prone to attack. Sometimes it is practically not possible to keep all of the machines up to date. Some networks consist of thousands of hosts and even with the help of automation applying frequent security, patches is still a challenging task. Organizations sometimes keep a strict security policy because tightening it prevents legal users from using the network. For example, an organization may give its employees remote access to the network or make a decision to keep an FTP server open.

Figure 2 shows an example of enterprise network. The network includes three distinct subnets: a DMZ (Demilitarized Zone), an EMS (Energy Management System) subnet, an internal subnet. An EMS subnet is a control-system network for power grids. It is assumed that host-grouping was already applied, based on analogous configurations; the workstation node can be a grouping of say 100 workstation machines with setups that are alike. The VPN server and the web server are directly accessible from the Internet. The web server accesses the file server through the NFS file-sharing protocol The VPN server is allowed to access all hosts that are present in the internal subnet. Outside access to the EMS subnet can be from the Citrix server only. The attacker's goal is to obtain permission to run code on the communication server from which an attacker could send commands to physical facilities that can harm the critical infrastructures.

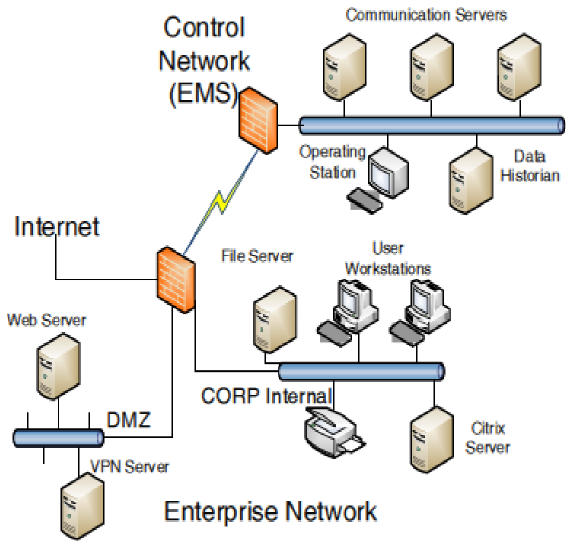


Figure 2: An example enterprise network

Attack graph provides an effective way to have clear understanding about the context and the relative importance of vulnerabilities in networks and systems. Attack graph analysis depends on complete and accurate model of the network. Network vulnerability scanners are used for creating such models. However, the scanning range has a fundamental limitation on the information available about the target host. Big networks typically employ several modes of connectivity and contain software packages and multiple files and platforms. Also network security is usually not enough to consider the presence or absence of isolated vulnerabilities.

2.3 Zero-Day Attacks

Leyla Bilge, Tudor Dumitra [5] suggests that existence of zero day attacks which exploit vulnerabilities and its duration is not completely known and is not disclosed publicly. Unfortunately, these serious threats are hard to find out, because data will not be present until the attack is discovered. Moreover, zero-day attacks are exceptional events that are generally not observed in lab experiments. The paper specifies a method for identifying zero-day attacks automatically from gathered data that starts recording when harmful binaries are downloaded on real hosts all around the world. When data set is searched for harmful files that exploit known vulnerabilities, it indicates which files appeared on the Internet before the vulnerabilities were disclosed. A zero-day attack is characterized by a vulnerability that is exploited in the wild before it is disclosed. Similarly, zero-day vulnerability is vulnerability employed in a zero-day attack. The aim is to measure the prevalence and duration of zero-day attacks and to compare the impact of zero-day vulnerabilities.

Vulnerabilities which are assigned with a CVE identifier are considered. In some cases vendors learn about vulnerability before it is exploited, but does not consider it high priority, and Cyber criminals may also prolong the release of exploits until they get a suitable target, to stop the discovery of the vulnerability. While the CVE database sometimes denotes when vendors were told about the vulnerabilities, it is generally not possible to estimate the exact date when the cyber criminals or the vendors discovered the vulnerability or which discovery came first. Hence the disclosure date of the vulnerability is called as "day zero" which symbolises the zero-day attack's end. In addition, some exploits are not employed for harmful activities before the disclosure date and are disseminated as proofs-of-concept in order to help the software vendor know about the vulnerability and the anti-virus vendors update their signatures. An opportunity is created for cyber criminals to create additional exploit, when disclosed vulnerabilities are not patched. A systematic study of zero-day attacks is done to identify a technique for analyzing zero-day attacks from the information available through the Worldwide Intelligence Network Environment (WINE). WINE is a platform for data intensive experiments in the field of cyber security. WINE includes field data collected by Symantec on 11 million hosts around the world. These hosts do not symbolize machines or honey pots in an artificial lab environment; they are real computers that are being targeted by cyber attacks. For instance, the binary reputation data set has information on binary executables downloaded by users. The anti- virus telemetry data set includes reports about host-based threats identified by Symantec's anti-virus products. The basic idea behind the technique is to find out executable files which are associated with exploits of known vulnerabilities.

When the vulnerabilities are undetected, the cyber criminals get a free pass to attack targets of their choice since they possess the knowledge about the new vulnerabilities. It is identified that 11 out of 18 vulnerabilities were not previously known to have been involved in zero-day attacks. A typical zero-day attack lasts 312 days on average and that, the volume of attacks increases rapidly after vulnerabilities are disclosed publicly. However, the serious threats are difficult to identify, because, in general, data is not available until after an attack is discovered. Zero-day attacks are more frequent and network vulnerabilities identified were not known zero-day vulnerabilities. It is difficult to be identified in serial connection.

2.4 Analysis on Software Vulnerability Life Cycles

Muhammad Shahzad, M. Zubair Shafiq, Alex X. Liu suggests that software systems inherently consist of vulnerabilities that have been exploited in the past which results in significant revenue losses. The study about the vulnerability life cycles can help in the various phases like development and maintenance. Designing of future security policies and conducting audits of past incidents are being carried out. In addition to that, such an analysis can help customers to estimate the security risks associated with different vendor's software product. The paper has an exploratory measurement study of a large software vulnerability data set containing numerous vulnerabilities disclosed since 1988[6]. Investigations about vulnerabilities are carried out in following seven dimensions:
(1) Phases in the vulnerabilities life cycles
(2) Evolution of vulnerabilities
(3) Functionalities of vulnerabilities
(4) Access requirement to exploit the vulnerabilities
(5) Level of risks in vulnerabilities
(6) Software products and
(7) Software vendors.

A large software vulnerability data set from two vulnerability repositories is taken into account:
(a) National Vulnerability Database (NVD),
(b) Open Source Vulnerability Database (OSVDB)
To systematically analyze patterns in our vulnerability data set, association rule is utilized to extract rules that represent exploitation behaviour of hackers and the patching behaviour of vendors. The most primitive and most exploited form of vulnerabilities is DOS, BO, and EXE; however, SQL, XSS, and PHP have also become significantly large. It is also observed that the percentage of remotely exploitable vulnerabilities have gradually increased to over 80% of all the vulnerabilities. Since 2008, the vendors have been becoming more agile in patching

the vulnerabilities and the access complexity of vulnerabilities has been increasing. However, even then, the average time taken by hackers to exploit vulnerability is smaller than that taken by the Patching of vulnerabilities in closed-source software is faster compared to open-source software and at the same time the exploitation is slower.
The exploratory analysis of vulnerability life cycles can discover interesting patterns for software vendors and software products which are useful in following ways:
1) A thorough analysis is useful in the software development processes.
2) Such analysis is helpful to develop the security policies that can handle future threats and attacks in a more effective manner.
3) An exploratory analysis gives insights about the former security incidents that are useful in their audit.
The exploratory analysis has significant findings that have a vital role to play in software development and deployment. However, the evolution of life cycle of various vulnerabilities is not fully analyzed.

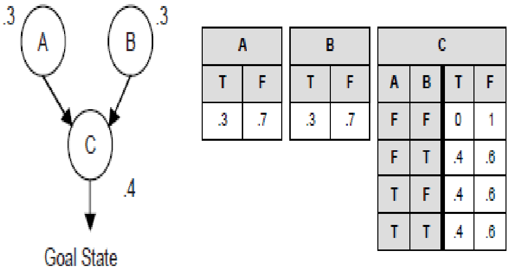
2.5 Quantified Security- A Weak Hypothesis

Vilhelm *et al.* [7] critically surveys previous work on analysis of security and quantitative representation. Such quantified security is represented as a general approach to precisely control and assess security. With respect to security perspective, underlying assumptions, types of validation and target of qualification, a significant part of work from 1981 to 2008 is being classified. In spite of applying a numerous techniques from fields such as economics, computer science and reliability theory to the problem, it is still unclear about what valid results exist with respect to operational security. Lack of validation makes quantified security a weak hypothesis. In addition to that, many assumptions in formal treatments are not empirically supported in operational security as they are taken from other fields. A number of risks are present as it depends on quantitative methods with validation. To make the concept of a weak hypothesis more precise, Karl Popper's model of scientific knowledge is taken into account particularly in the empirical sciences. Descriptive (e.g. quantitative) methods that attempt to represent empirical facts are seen as hypotheses that can be judged as either incorrect or correct to a specified degree. While many methods of generating hypotheses are important, the only way to learn about the correctness of a hypothesis is by challenging empirical tests. If hypotheses succeed in describing outcomes of experiments, e.g. by repetitively including additional tests, they get collaborated. Alternatively, hypotheses of the statement can also be faked by inconsistent anomalies and evidences. The latter requires replacing or modification of hypotheses.

A hypothesis where too little is known about its correctness to call it corroborated or falsified is considered. They ultimately depend on a lack of empirical tests or unclear evidence. Measurements of security, metrics and models are here connected to each other in the following way. A measurement is made by noticing the outcome of an event using some suitable method to collect the result data. A metric assigns such information onto some kind of scale in order to accurately denote several security features of a system. For valid assignments, correct and sharp assessment and comparison of systems, the absolute values or number ordering is used. Furthermore, the idea of a security model in the scheme is to stipulate a formal representation (e.g. sets of derivations) that corresponds well to security for systems under consideration. A valid model can then be used to obtain quantities of interest using suitable data and parameters. Models are needed for quantification at once when there is a nontrivial correlation between possible measurements and the feature that one wants to quantify: data from potentially imprecise or uncertain measurements needs to be related to some definition of security. Building and validating quantitative models for security is of difficult interest when one needs or claims accuracy in describing security. Attackers have the capacity to decide and exhibit series of attacks that are highly correlated. However, there are limitations such as security failures which are being highly directed. The outcome shows that the validity of most methods is unclear.

2.6 Network Security Evaluation Using Bayesian Networks

A **Dynamic Bayesian Network** (DBN) is a Bayesian Network that relates variables to one another over time steps that are adjacent. DBN is often called a Two-Time slice BN (2TBN) because the value of a variable can be computed from the immediate prior value (time T-1) and the internal regresses at any point in time T. DBNs were in the early 1990s developed by Paul Dagum when he led research funded by two National Science Foundation grants at Stanford University's Section on Medical Informatics. DBN's were developed to unify and extrapolate traditional linear state-space models such as normal and linear forecasting models such as ARMA, kalman filters and simple dependency models such as hidden Markov models into a generalized probabilistic representation and inference mechanism for arbitrary non-normal, non-linear and time-dependent domains. **Marcel *et al.*** [8] proposes that the overall security of networked information systems should be measured and improved when the increasing dependence is being given. The current security metrics focuses on computing individual vulnerabilities without taking into account their combined effects. The previous works handles the issue by determining the causal relationships between vulnerabilities. However, the emerging nature of vulnerabilities and networks are ignored. A Dynamic Bayesian Networks based model is proposed to include temporal factors such as the availability of exploit patches. To demonstrate the potential applications, a model is proposed and it is implemented in two different case studies. For continuously measuring network security in a dynamic environment, the novel model provides a clear view on theoretical foundation and practical framework.



A	
T	F
.3	.7

B	
T	F
.3	.7

C			
A	B	T	F
F	F	0	1
F	T	.4	.6
T	F	.4	.6
T	T	.4	.6

Figure 3: Representing Attack Graphs as BN's

Figure 3 show the attack graphs which is being represented as Bayesian Networks. Attack graphs emphasize the knowledge about how multiple vulnerabilities may be joined for an attack to occur. The model provides system states using security conditions, such as the existence of vulnerabilities on the host. The limitation pointed out here is the lack of consideration for temporal factors in previous works on evaluating network security.

2.7 An Attack Graph-Based Probabilistic Security Metric

To protect critical resources in the present network environments, it is desirable to quantify the possibility of potential multi-step attacks along with multiple vulnerabilities. The model of causal relationships between vulnerabilities such as the attack graph now makes it more sensible. **Lingyu *et al.*** [9] proposes an attack graph-based probabilistic metric for securing networks and determines its efficient evaluation. The basic metric is first defined and meaningful interpretation is provided to the metric. The definition in more complex attack graphs with cycles is studied and the definition is extended accordingly. The metric computed directly from its definition is not effective in many cases and hence heuristics are proposed to improve the effectiveness of such evaluation.

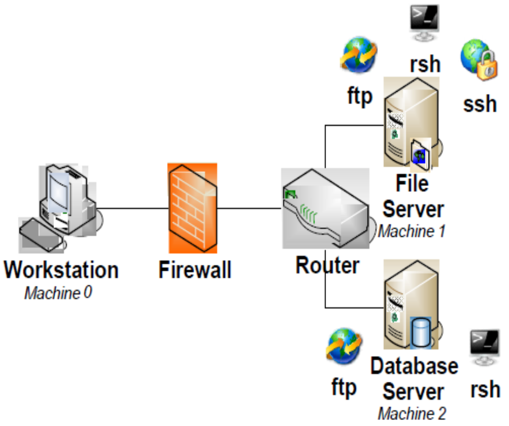


Figure 4: An Example of Network Configuration

Attack graphs show how multiple vulnerabilities are joined for dealing with an intrusion. Security-related conditions are represented using the system state, and an exploit of vulnerabilities between connected hosts is taken as a transition between system states. Figure 4 shows that Machine 1 is a file server behind the firewall that provides secure shell (ssh), file transfer (ftp), and remote shell (rsh) services. Machine 2 is an internal database server that offers rsh and ftp services. The firewall allows ssh, ftp and rsh traffic to both servers and stops the other incoming traffic. The causal relationships between vulnerabilities are clearly understood and usually encoded as attack graphs.

Attack graphs help to understand whether given critical resources can strike a balance through multi-step attacks. Clearly, there is a gap between existing security metrics and qualitative models of vulnerabilities, which are usually limited to binary views of security. A probabilistic metric is proposed for measuring network security. The metric draws strength from both attack graph model and the existing security metrics. More specifically, the measurements of individual vulnerabilities obtained from existing metrics are combined into an overall score of the network. The combination is based on the relationships that are encoded in an attack graph. The key challenge lies in dealing complex attack graphs with cycles. The basic metric is defined without considering cycles. An intuitive interpretation of the metric is provided. Based on the interpretation, the definition is extended to attack graphs with cycles. Finally, the efficient computation of the metric is analysed.

The advantage of using an attack graph based probabilistic security metric is that it gains strength from both attack graph model and existing security metrics. However, most of the vulnerabilities even after being discovered still remain in the network due to cost factors, mission factors and environmental factors. Also it focuses only on and qualitative models of vulnerabilities and individual vulnerabilities which are generally restricted to binary views of security.

2.8 Attack Graph Generation

It is necessary to consider multi-stage and multi-host attacks while evaluating the security of an enterprise network. A determined attacker tries to penetrate deeper into the network by switching from one machine to another. Hence, configuring an enterprise network in a secure manner is a challenging task for human beings. There are many possible interactions among multiple components and hosts in a network, such that the configuration of one machine will have effect on the security of other machines in the network. Hence, it is important to design automated tools that can deal with the configuration of an enterprise network and determine possible security vulnerabilities. Such a tool will be of no use if it cannot intimate a system administrator with elaborate information about the identified problems. An attack graph that illustrates all possible multi-stage and multi- host attack paths is difficult for a systemadministrator to understand the behaviour of the threats and decide upon appropriate solutions.

Xinming Ou *et al.* [10] suggests a new method to represent and produce attack graphs. They are essential for examining security susceptibilities in enterprise networks. A logical attack graph permanently has size polynomial to the network that is being investigated. Their attack graph generation tool is built upon MulVAL, a network security analyzer based on logical programming. A derivation trace is produced in the MulVAL logic programming engine, and how to practise in order to obtain a logical attack graph within the given quadratic time. The authors also demonstrated experimental proof that their logical attack graph generation algorithm is very competent in the paper. Logical attack graphs, which directly describes logical dependencies among configuration information and attack goals. The growth trend for MulVAL is not obvious as the running time is too short.

2.9 Minimum-cost Network Hardening

Some vulnerability may look like acceptable risks when considered in isolation point of view when defending one's network against the cyber attack. However an intruder can often infiltrate a well-protected network through a multi-step intrusion, in which each step can be used for the next consecutively. Attack graphs can disclose the threat by calculating possible sequences of exploits that can be used to compromise the given critical resources. However, to remove the threat, attack graphs do not directly give a solution. Obtaining a solution by hand is error-prone and hard, particularly for larger and less secure networks because their attack graphs are very complicated. The paper proposes a solution to automate the task of network hardening against multi-step intrusions. Unlike existing approaches, our solution is consists of initially satisfied conditions alone because they can be independently disabled. Given critical resources are represented as a logic proposition of initial conditions and then the proposition is simplified to make hardening options explicit. Finally solutions with minimum cost are chosen.

Lingyu *et al.* [11] includes a framework of an improved one-pass algorithm and minimum network hardening problem for the logic proposition. It goes beyond the attack paths to evaluate actual sets of network hardening measures (initial network conditions- assignment) which guarantee the security of given critical resources. However considering vulnerabilities in isolation is insufficient in the analysis of networks that are vulnerable to attacks. It is because attackers often combine exploits against multiple vulnerabilities to reach their objective.

2.10 Risk of using Vulnerable Components

To estimate the risks to which the system is exposed, data about the architecture and the vulnerabilities affecting a distributed system can be used. The aim of risk assessment is to evaluate the likelihood that identifiable vulnerabilities will harm, weighting their existence with the loss they may cause. **Davide *et al*.** [12] provides a tool that helps in decision making to conclude if the extra cost of a more secure component is worth to be offered. To measure the risk, a quantitative approach is proposed based on the knowledge of:
–The vulnerabilities of links and components and a measure of their exploitability.
– The dependencies that the architecture of the system generates among vulnerabilities, since vulnerability can be exploited more easily.
– The possible attacks against the system.
To estimate how much one should believe in system trustworthiness, risk evaluation is used.
–To perform comparison between different solutions.
Designer often have the choice of using different architectural choices and different components. Quantitative risk assessment is the key in order to provide a tool for making a decision if extra cost of a more secure component is worth to be offered.

The concept of risk evaluation can be used to check out how much belief should one have on system trustworthiness and perform comparison of different solutions. In fact designers have often the option of using different components. The limitations are most systems seem to be vulnerable in one way or the other, but it does not mean that the system is too flawed to use. Analysis of risk in large distributed systems is still a tough problem for security managers since it requires an accurate balance of experience and skill. All the attacks are not discovered successfully.

3. CONCLUSION

The ultimate strength of the dependability concept depends on its various attributes such as availability, integrity, maintainability, confidentiality, reliability, safety and security. It is not simply an inherent property of the system but a situated concept with accompanying work practices. Some of the important concepts and metrics that are effectively used in securing computer networks and measuring the risk of vulnerabilities along with their advantages and limitations are being discussed. The literature review would turn up the attention to deal with further improvements by proposing new network security metrics for measuring the risks of unknown vulnerabilities and computing metric scores for zero day vulnerabilities.

REFERENCES

[1] B.Kalpana 'A Literature Review On Dependable And Secure Computing', International Journal of Research in Computer Applications and Robotics-IJRCAR, Vol. 2, Issue No. 8, PP. 19-25.

[2] Lingyu Wang, Member, IEEE, Sushil Jajodia, Fellow, IEEE,Anoop Singhal, Senior Member, IEEE, Pengsu Cheng, and Steven Noel, Member, IEEE," k-Zero Day Safety: A Network Security Metric for Measuring the Risk of Unknown Vulnerabilities", IEEE Transactions On Dependable And Secure Computing, Vol. 11, PP 31-44.

[3] Patil Priyanka Nagnath, Prakash B.Dhainje, Deshmuk pradeep K , "Attack Graph-Based Security Metrics and other metrics for producing Security to the Computer Network," International Journal of Comp. Science and Information Technologies(IJCSIT '15), Vol.6(2), PP 941-943.

[4] Tito Waluyo Purboyo, Kuspriyanto, "A Review of Network Security Metrics," International Journal of Advanced Research in Comp. Science and Software Eng, Vol 3, Issue No. 9.

[5] Leyla Bilge,Tudor Dumitras, "An Empirical Study of Zero-Day Attacks in the Real world," Proc. Hawaii Int'l Conf. System Sciences, pp. 1-12.

[6] M. Shahzad, M. Shafiq, and A. Liu, "A Large Scale Exploratory Analysis of Software Vulnerability Life Cycles," Proc. 34th Int'l Conf. Software Eng. (ICSE '12).

[7] V. Verendel, "Quantified Security Is a Weak Hypothesis: A Critical Survey of Results and Assumptions," Proc. Workshop New Security Paradigms Workshop (NSPW '09), pp. 37-50.

[8] M. Frigault, L. Wang, A. Singhal, and S. Jajodia, "Measuring Network Security Using Dynamic Bayesian Network," Proc.Fourth ACM Workshop Quality of Protection (QoP '08).

[9] L. Wang, T. Islam, T. Long, A. Singhal, and S. Jajodia, "An Attack Graph-Based Probabilistic Security Metric," Proc. 22nd Ann. IFIP WG 11.3 Working Conf. Data and Applications Security.

[10] X. Ou, W. Boyer, and M. McQueen, "A Scalable Approach to Attack Graph Generation," Proc. 13th ACM Conf. Computer Comm. Security (CCS' 06), pp. 336-345.

[11] L. Wang, S. Noel, and S. Jajodia, "Minimum-Cost Network Hardening Using Attack Graphs," Computer Comm., vol. 29,no. 18, pp. 3812-3824.

[12] D. Balzarotti, M. Monga, and S. Sicari, "Assessing the Risk of Using Vulnerable Components," Proc. ACM Second Workshop Quality of Protection (QoP '05), pp. 65-78.