IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p- ISSN: 2278-8727Volume 16, Issue 3, Ver. IX (May-Jun. 2014), PP 06-12
www.iosrjournals.org

A Short-Normalized Attack Graph Based Approach for Network Attack Analysis

Gouri R Patil[1], A. Damodaram[2]
[1]*Research Scholar, MJCET, JNTU university, Hyderabad, India.*
[2]*Professsor, JNTU university, Hyderabad, India.*

Abstract: *Attack graphs are the graphs which describe attack scenarios, play important roles in analyzing network threats. These attack graphs are able to reveal such potential threats by evaluating the all possible sequences that an attacker can follow to compromise given critical resources or nodes. An Attack graph specifies an attack scenario that results in compromising network values. There are so many methods proposed to evaluate the network security in attack graphs. But no method specifies the overhead occurred due to the evaluation of network security at each and every node. This paper addresses the problem of overhead occurred due to the network security evaluation in Short-Normalizedattack graphs by evaluating a factor called network security risk. In this paper first the possible n valid attack paths are going to be calculated and then the security risk is going to be calculated for those n valid paths. This security risk denotes the amount of overhead occurred due to this evaluation.*

Keywords: *Network Security,Network Configuration, Short-Normalized Attack Graphs, Security Risk Evaluation,*

I. Introduction

Now a day's the society has become increasingly dependant on the proper functioning and reliability of a large number of interconnected information transfer systems though the major issue is to secure such systems, it is necessary to evaluate the amount of security provided by various network configurations. Thus it is important to design an automatic tool that can analyze the configuration of a network and find security threats and the attack paths. In a network with significant resources, certain threats may seem to be insignificant for some operations. An attacker may take advantage of it and make use of sequences related to those threats. Attack graphs are the graphs which describe attack scenarios, play important roles in analyzing network threats. These attack graphs are able to reveal such potential threats by evaluating the all possible sequences that an attacker can follow to compromise given critical resources or nodes. An Attack graph specifies an attack scenario that results in compromising network values. It tells us how an attacker gains access to the victim; how and which vulnerability attacker can take advantage of and what kind of damage may be done that can impact the network. Attack graphs provide the complete information about the network thus provides the security from attackers.

Basically networks are designed to perform various information transfer scenarios. These networks tend to get the maximum amount of quality of a data (text, audio,and video) on reception. The various network modeling scenarios are going to be done by aiming this concept. Butthe performance of the network is getting degraded when the data passing through the overall nodes in a network is suffered from attacks by various attackers at various instants. Attack graphs are able to specify attack scenarios, play important role in analyzing network threats. This evaluation is becoming an overhead to the network. I.e. analyzing each and every node wherethere is a possibility to attack and mentioning precautionary measures to overcome it, thus providing security in networks.

There are so many methods proposed on attack graphs [1] [2]. They have been leveraged to evaluate network security, and some further models and examples on network security metric [3][4][5][6][7] have been constructed. Firstly, previous works encounter the scalability problem in the generation of the attack graphs, with the size of the attack graphs increasing exponentially with that of the network. Although some researchers try to address this problem [5], their result graphs are still too large and complicated to be analyzed efficiently.

Secondly, most of previous attack graphs are designed for a single target, and can not be used to evaluate overall security of networks with several targets. While managing a typical network including multiple critical resources, network administrators would like Corresponding author. To evaluate those resources as a whole rather than reporting each one separately.Thirdly, it is easy to describe outside attackers' threat, but few suggestions have been described to prevent inside malicious attackers from attacking networks. In [8] a new type of attack graph model to detect an intrusion is proposed called as MP (Multiple-prerequisite) attack graphs. Multiple-prerequisite graph (MP graph) is a type of attack graph that has been developed to help defending large scale enterprise network. In this model two stochastic models are mentioned for quantitative security evaluation. These models are constructed based on the use of Markov Decision Process to model the attacker's behaviors. But the problem associated with this model, it is able to evaluate the network security under static conditions only. I.e. there is no any information about the varying conditions of network structure and network content. [9] proposed a solution to this problem by considering the dynamic characteristics of the nodes in the network but it didn't give any analysis about the overhead occurred during this analysis of network security.But the problem associated with this approach, the overhead occurring due to the evaluation of cost for these attacks is increased due to dynamic conditions of network.

To solve the above problem in this paper an efficient network security evaluation is performed on Short-Normalized attack graphs. In this paper, based on the *n*-valid paths, we present an algorithm to find outthe security risk during finding the total attack paths. Our method can be applicable to practical attack graphs of enterprise networks with thousands of hosts.

The rest of the paper is organized as follows: section II gives the information about the attack graph model used in this paper. Section III describes the main problem occurred with some illustrations. Section IV gives the evaluations for valid attack paths and the security risk occurred due to that evaluation. The performance evaluation is discussed in section V with an example. This section also gives the comparison results. Finally conclusions are illustrated in SectionVI.

II. Attack Graph Model

A Short-Normalized attack graph can generally be represented as a directed graph with two types of nodes, exploits and security conditions (or simply conditions). We then formally define Short-Normalized attack graphs as follows:

Definition 1: Let AP is a set of atomic propositions, $C_0$and $C_r$is a set of initial and reachable conditions, respectively. L: $C_0 \cup C_r \rightarrow AP$is a labeling function of conditions with a true proposition. Let Tbe a set of exploits, and $E \subset (T \times Cr) \cup (C0 \cup Cr) \times T$ is a set of sides between nodes (conditions or exploits). An attack graph is a tuple $AG = (C0 \cup Cr, T, E, L)$.

To facilitate understanding the attack graph, it is convenient to interpret an attack graph as a simple logic program as follows. Each condition in the attack graph is regarded as a logic variable. The interdependency between exploits and conditions now becomes logic propositions involving the two connectives AND and OR. AND or OR means both or one of the conditions are required by each exploit, respectively.

Property 1: For every exploit node $\tau \in T$, let Pre(τ) be the set of τ's pre-conditions and Post(τ) be the set of itspost-condition, $(\wedge L(ci) \rightarrow L(cj))$, where $c_i \in$Pre(τ),$cj \in$Pre(τ),that shows when all the pre-conditions of exploitτ aretrue, the post-conditions of exploit τ will be true.

Figure 1 shows a simple example of attack graphs,where $C_0 = \{c1,c2,c3\}$, $T = \{\tau\ 1, \tau 2, \tau 3\}$, $C_r = \{c_4, c_5, c_6, c_7\}$, $C_f = \{c_7\}$. Attackers can arrive at $c_5$by theexploit $\tau_1$and make $c_4$true through τ_2. When the conditions $c_5$and $c_4$are both true, they further make thecrucial condition $c_7$true. Hence, $\perp \rightarrow \tau 1 \rightarrow \tau 2 \rightarrow \tau 3$ is anattack path to c_7.

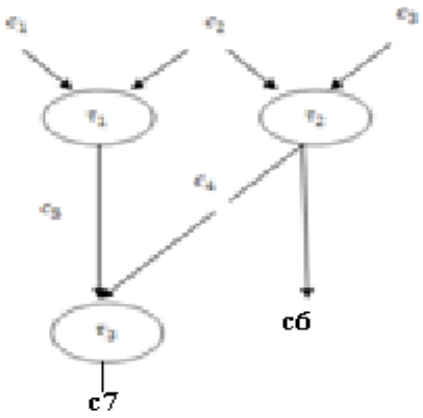

Fig1: A simple example of Attack graph

III. Problem Formulation

By traversing all the possible sequences of exploits to critical resources in the attack graphs, we can obtain all the possible attacks. Hence, attack graphs reveal the potential threats. Attack graphs, however, are often so complex that one user is difficult to comprehend it fully and reach appropriate configuration decisions. It has been demonstrated in the well-known example in the study of attack graphs. In this application example, the attacker's machine is denoted machine 0, and the two victim machines are denoted 1 and 2, respectively. The details of the attack scenario (such as network topology, available services, operating systems, etc.) are not needed here.

Figure 2 shows the attack graph for the example. In the figure, exploits are denoted as ovals, and conditions as plain text. Numbers in parenthesis identify associated machines. For example, root(2) meansattackers own root privilege on machine 2, and rsh(2,1)denotes the execution of the *rsh*exploit from machine 2to machine 1. More details can be referred to [3,4].

If we focus on the set of the crucial conditions *Cf*={user(1)}, the sequence of exploits Path1= $\omega \rightarrow$host(0) $\rightarrow$host(1) is one attack path of *Cf*, thatshows the attacker can first establish a trust relationshipfrom his machine (host 0) to host 1 (the condition trust(1,0)) . Path2=$\omega \rightarrow$host(0)$\rightarrow$host(1)$\rightarrow$host(1,2)$\rightarrow$host(2,1) is also one attack path of *Cf*. But this attack path has a loop. In realistic attack scenarios, the attacker generally does not choose this attack path for he does not make an effort in owned privilege. Furthermore, although attack graphs lay out all the theoretical attack paths, those attack paths with long distance practically cannot be used by attackers. Hence, we define a*n-valid attack path* as a non-loop attack path with the distance less than *n*. Further, we desire to answer the following questions.

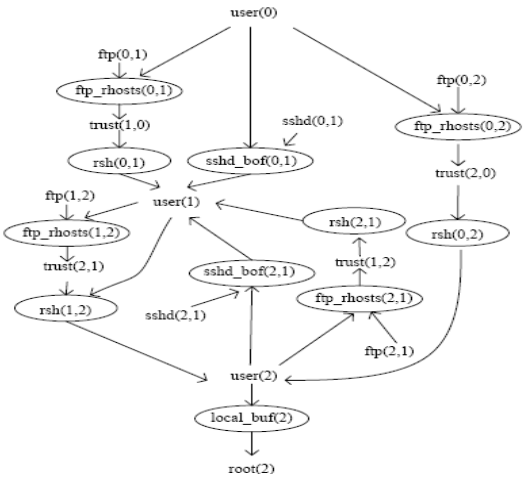

Fig2: The attack graph for above example

Question 1: Given the set of crucial conditions *Cf*and*n*, what are all the *n*-valid attack paths to *Cf*?In the set of attack paths to the crucial conditions setC_f^1={user(1)}, the shortest attack path is *Path*3=host(0,1) with one exploit . However, in the set ofattack paths to the crucial conditions set C_f^2={user(2)}, the shortest attack path is *Path*4=sshd_bof(0,1)→Ftp_rhost(1,2)→rsh(1,2) with three exploits. Obviously, $C_f^1$and C_f^2 are suffering from different security risk.

Thus, the second question we should answer is:

Question 2: Given the set of crucial conditions *Cf*, howto measure the security risk of *Cf*?Removing different vulnerabilities usually incursdifferent costs, and in practice removing all knownvulnerabilities is typically impractical due to lack ofpatches or upgrades and the incurred cost.

IV. Design Approach

A. Computing n-Valid Attack Paths

In this section, we discuss how to obtain all the *n*-validattack paths used by attacker to compromise the given set of crucial conditions *Cf*. Theorem 1 follows directly fromthe definitions.

Theorem 1Given the constant *n* and an attack path to*Cf*, *Path*=$\omega\rightarrow \tau1\rightarrow \tau2\rightarrow\ldots\rightarrow \tau l$, *l*<*n*. p*ath*is a *n*-validattack path if and only if $\forall\ \tau i(1\le i<l)$,$pre(\tau i)\cap(\cup_{j-i-1}^{l})post(\tau i))=\varnothing$.

In other words, for any exploit τi in an *n*-valid attackpath, the pre-condition of τi is not the post-condition ofthe succeeding exploits of τi.Through further studying general attack graphs, wefind that there exist some *n*–valid attack paths, whichindicate the same exploits dependency relation. We canexplain this by Figure 1, where existing two 4-validattack paths $\omega\rightarrow \tau1\rightarrow \tau2\rightarrow\tau3$ and $\omega\rightarrow \tau2\rightarrow \tau1\rightarrow \tau3$. Hence,the exploit τ3 depends on both τ1 and τ2. Nevertheless,there does not exist dependency relation between τ1 andτ2. We consider these attack paths are equivalent to eachother.To obtain all the *n*-valid attack paths to *Cf*, we definetwo kinds of sets. Let *AG*= (*C0*∪*Cr*,*T*,*E*,*L*) be an attackgraph. For each condition *c*∈*Cr*, letPATHS(*c*)={$Path^{(i)}(c)$} denote the set of reachable attackpaths to *c*, where $Path^{(i)}(c)$= $\omega\rightarrow \tau1\rightarrow \tau2\rightarrow\ldots\rightarrow \tau l$with*c*∈Post(τ*l*) is the *i*th reachable attack path to *c*. Similarly,PATHS(τ)={ $Path^{(j)}(\tau)$} denotes all the reachable attackpaths to exploit $\tau\in T$, where $Path^{(j)}(\tau)=\omega\rightarrow \tau1\rightarrow \tau2\rightarrow\ldots\rightarrow\tau m$means the *j*th reachable attack pathto τ=τ*m*.We then consider the interrelations between thereachable attack paths of exploits and conditions. Let*c*∈*Cr* be the reachable condition and τi be the exploitpointing to *c*, where $1\le i\le m$, each reachable attack path ofτ*i*is obviously the reachable attack path of *c*, thusPATHS(*c*)=(∪PATHS(τ*i*)).Let τ∈*T* be the exploit and *cj*be the condition pointingtoτ, where $1\le i\le n$. the reachable attack path of τ is*path*→τ, where *path*∈PATHS(*c*1)⊕…⊕PATHS(*cn*).

B. Measuring Security risk

According to the above discussion, we may use theprocedure *obtain_paths*(*g*,*n*) to compute all the *n*-validattack paths to the given set of crucial conditions *Cf*.. Inthis section, we will measure the security risk of thecrucial conditions *Cf*using attack graphs.Let PATHS(*Cf*) ={*Path*(*i*)|$1\le i\le m$ } be all the attackpaths to *Cf*, the security risk of *Cf*depend on three factors.The first is the number of attack paths to *Cf*, denoted as *m*,and the more attack paths means there are moreopportunity for attacker to compromise crucial conditions.

The second is the distance of attack path *Path*(*i*), denotedas *li* (*li*<*n*), and the longer distance of attack paths impliesthe attacker should have greater endurance to reach theattack goals. The third is the number of kinds of exploitsin PATHS(*Cf*), denoted as *k*, and the more kinds ofexploits indicate that attacker needs have moreknowledge on different exploit technologies. Therefore,we define the security risk of *Cf*as the following formula:

$$\text{Risk}=\frac{1}{k}w+(1-w)\sum_{i=1}^{m}\frac{1}{l_i} \qquad (1)$$

Where*w* and 1-*w* are the weight of the factors forattacker's endurance and knowledge, respectively.

V. Performance Analysis And Experimental Results

To justify our approach to analyzing the network security, we apply the approach to a well-known example in figure 2.

(1) Computing*5*-valid attack paths

Let *Cf*={user(2)} and *n*=5. We use the procedure *obtain_paths*to compute all the *5*-valid attack paths to *Cf*shown in the following:

*Path*1=*ω*→Ftp_rhost(0,2) →rsh(0,2),

*Path*2=*ω*→Ftp_rhost(0,1)→rsh(0,1)→Ftp_rhost(1,2)→rsh(1,2),

*Path*3=*ω*→sshd_bof(0,1)→Ftp_rhost(1,2)

→rsh(1,2),

(2) Measuring Security Risk

From these attack paths, we know *l*1=2, *l*2=4, *l*3=3, *m*=3 and *k*=3 for attacker need have knowledge on Ftp_rhost, rsh and sshd_bof exploit technologies to reach the crucial condition user(1) and user(2). Supposing *w*=0.5, we use formula 1 to compute the security risk of {user (2)}, i.e.

$$Risk=(1/3).5+(1-.5)((1/2)+(1/3)+(1/4))=.71$$

Thefollowing figures give the complete illustration about the proposed method.

The figure 3 represents the basic node distribution in a random network topology

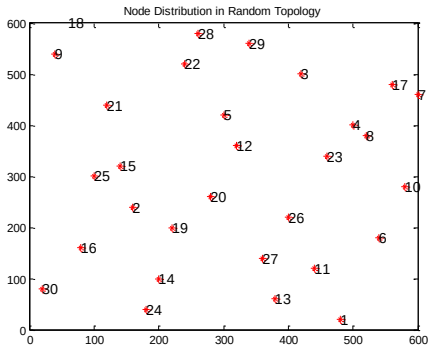

Fig 3: Node distribution in a random topology

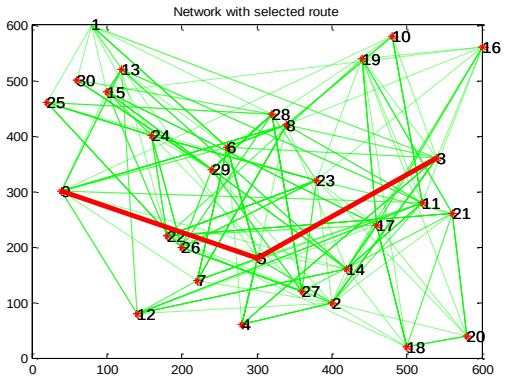

Fig4: A generated attack graph for above node distribution

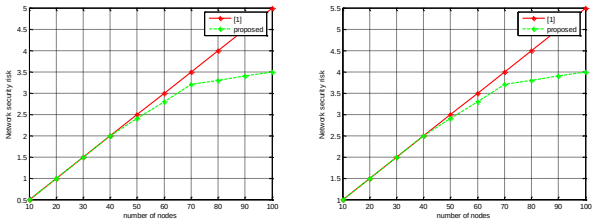

Fig5: node distribution versus network security risk for Attack path1 and Attack path2

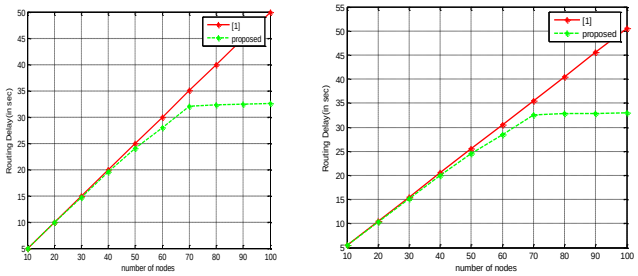

Fig6: node distribution versus routing delay for Attack path1 and Attack path2

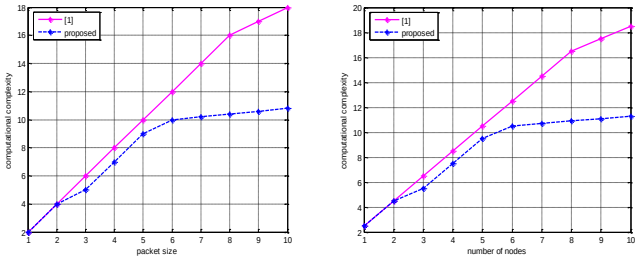

Fig7: Node distribution versus network security risk for Attack path1 and Attack path2

The results presented above illustrates that the proposed method is succeeded in evaluating risk and overhead over attack graphs. The conclusions are mentioned in next section

VI. Conclusions

In this paper, a new approach was proposed to compute non-loop attack paths with the given distance in the Short-Normalized attack graphs. We further present the novel approach to measuring the security risk. The approach can analyze attack graphs for defending network security in polynomial time. The experiments show it is of good scalability for large enterprise networks with thousands of hosts, which have complex Short-Normalized attack graphs.

References

[1]. C. A. Phillips and L. P. Swiler, "A graph-based system for network vulnerability analysis," in Workshop on New Security Paradigms, 1998, pp. 71–79.

[2]. O. Sheyner, J. W. Haines, S. Jha, R. Lippmann, and J. M. Wing, "Automated generation and analysis of attack graphs," in IEEE Symposium on Security and Privacy, 2002, pp. 273–284.

[3]. L. Wang, A. Singhal, and S. Jajodia, "Toward measuring network security using attack graphs," in QoP, G. Karjoth and K. Stølen, Eds. ACM, 2007, pp. 49–54.

[4]. S. Noel, S. Jajodia, B. O'Berry, and M. Jacobs, "Efficient minimum-cost network hardening via exploit dependency graphs," in ACSAC. IEEE Computer Society, 2003, pp. 86–95.

[5]. L. Wang, T. Islam, T. Long, A. Singhal, and S. Jajodia, "An attack graph-based probabilistic security metric," in DBSec, ser. Lecture Notes in Computer Science, V. Atluri, Ed., vol. 5094. Springer, 2008, pp. 283–296.

[6]. J. Pamula, S. Jajodia, P. Ammann, and V. Swarup, "A weakest-adversary security metric for network configuration security analysis," in QoP, G. Karjoth and F. Massacci, Eds. ACM, 2006, pp. 31–38.

[7]. M. Frigault, L. Wang, A. Singhal, and S. Jajodia, "Measuring network security using dynamic bayesian network," in QoP, A. Ozment and K. Stølen, Eds. ACM, 2008, pp. 23–30.

[8]. Feinberg, E. A. and Shwartz, A., Handbook of Markov decision process,Kluwer Academic Publishers, Boston, 2001
[9]. Wang Liu, Kejie Lu, Jianping Wang,Liusheng Huang, and Dapeng Oliver Wu, "Onthe Throughput Capacity of Wireless SensorNetworks with Mobile Relays", IEEETransactions on Vehicular Technology, VOL.61, NO. 4, MAY 2012.