A New Risk Assessment Framework Using Graph Theory for Complex ICT Systems

M.Yassine Naghmouchi
Orange Labs, France
mohamedyassine.naghmouchi@orange.com

Nancy Perrot
Orange labs, France
nancy.perrot@orange.com

Nizar Kheir
Thales Group, France
nizar.kheir@thalesgroup.com

A.Ridha Mahjoub
Université Paris-Dauphine
PSL Research University
CNRS, LAMSADE
75016, Paris, France
mahjoub@lamsade.dauphine.fr

Jean-Philippe Wary
Orange labs, France
jeanphilippe.wary@orange.com

ABSTRACT

In this paper, we propose a new risk analysis framework that enables to supervise risks in complex and distributed systems. Our contribution is twofold. First, we provide the Risk Assessment Graphs (RAGs) as a model of risk analysis. This graph-based model is adaptable to the system changes over the time. We also introduce the potentiality and the accessibility functions which, during each time slot, evaluate respectively the chance of exploiting the RAG's nodes, and the connection time between these nodes. In addition, we provide a worst-case risk evaluation approach, based on the assumption that the intruder threats usually aim at maximising their benefits by inflicting the maximum damage to the target system (i.e. choosing the most likely paths in the RAG). We then introduce three security metrics: the propagated risk, the node risk and the global risk. We illustrate the use of our framework through the simple example of an enterprise email service. Our framework achieves both flexibility and generality requirements, it can be used to assess the external threats as well as the insider ones, and it applies to a wide set of applications.

Keywords

Risk assessment; Graph Theory; Complex ICT Systems

1. INTRODUCTION

In the past, information systems were statically designed, and almost did not evolve during runtime, whereas current systems are complex. In fact, they are composed of a large number of heterogeneous elements, connected by non-linear interactions, often of different types (e.g. physical and virtual links). These systems are also subject to external and insider inferences (e.g. intruder threats). In addition, the elements of the system and the interactions between them evolve over time (e.g. the evolution of both the topology and its associated vulnerabilities). In the context of such complex systems, existing risk assessment methodologies, such as *Scoring methods* and *attack graphs*, suffer from multiple limitations.

Scoring methods [4], [2], [3] provide a common basis to share information about vulnerabilities and their severity. However, they only consider intrinsic properties of a security flaw without considering the way a vulnerable asset is integrated into a target system, its importance, and how it may affect other vulnerabilities. So that they cannot be used as a standalone metric to leverage risks in a target real-world system. To address this limitation, current approaches in the literature usually compose elementary vulnerabilities that may be identified in a target system, and their relationships, through a graph-based model. Nonetheless, current graph models also present some limitations.

Attack graphs [12], [11], [6], [9], [14] are used to assess the risks associated with system vulnerabilities. This kind of graphs highlights the cumulative effect of multiple consecutive attack steps, and each path in the graph leading to an undesirable state. One major limitation of attack graphs is the static representation of vulnerabilities and the underlying system topology. Furthermore, while the topology captured by attack graphs allows to describe the causal relationships between the system nodes, it is still insufficient to give a correct risk estimate. In fact, a risk may propagate from a source asset to a target one if a link between them exists. This risk could be higher if the couple of assets remains connected on this link for a longer time. Thus, the access duration is an important factor of risk.

These limitations present several challenges to defenders. A first challenge consists in modelling the different interactions between a large number of heterogeneous elements. To this end, a description of the system topology is needed. A second challenge consists in defining the adequate models taking into account the vulnerabilities, the topology, and their constant evolution over time. In particular, a metric indicating how much time the assets are interconnected, in a given period of time, is interesting as it enables a more fine grained assessment of risk propagations. This concept of connection time is called *the accessibility.*

In this paper, we propose a new risk assessment frame-

MIST'16, October 28 2016, Vienna, Austria
© 2016 ACM. ISBN 978-1-4503-4571-2/16/10…$15.00
DOI: http://dx.doi.org/10.1145/2995959.2995969

work to cope with the aforementioned challenges. Our contribution is twofold. First, based on graph theory [7], we introduce the concept of RAGs as a tool for dynamic risk analysis. These graphs properly characterize the complex systems by capturing both the topological accessibility features of the target system and security information in terms of vulnerabilities as well as their causal relationships. They take into account not only the current system state, but also the way it evolves throughout a period of time. In addition, all possible intruders, attack scenarios and their target assets are explicitly considered as paths in the RAGs. Moreover, we propose a new real-time risk evaluation approach. Our methodology is quantitative and based on a worst case scenario (i.e. *the most likely path*). Our framework achieves both flexibility and generality requirements. It can be applied to assess both external and insider intruders.

The rest of the paper is organized as follows. In Section 2, we present our RAGs model. In Section 3, we present our risk evaluation approach. In Section 4, we illustrate the use of our approach through the example of an email enterprise service. Some concluding remarks and indications for ongoing work are giving in Section 5.

2. THE RISK ASSESSMENT GRAPHS

In this section, we formally define the RAGs model. We define the security metrics used to evaluate the nodes, namely *the potentiality function* and *the impact*, and those used to evaluate the arcs, namely *the accessibility function.*

Let $I = \{1, \dots, T\}$ be a discrete time interval. We model the system by a set of directed graphs $(G_t = (V, A_t))_{t \in I}$. The set of nodes V is partitioned into two specified subsets U and W. The nodes in U represent *the access points.* These are special assets that may be used by insider or external intruders as entry points to attack the system.

Let Λ be the set of all system assets (except the access points). Let V_a be a set of all vulnerabilities of an asset $a \in \Lambda$. A node in W represents an asset-vulnerability pair where $w = (a, v)$, $a \in \Lambda$ and $v \in V_a$. Each node in W is evaluated by the potentiality and the impact which are defined as follows.

The potentiality function f of a node $w = (a, v)$ at time t is the probability for the vulnerability v to be exploited on a at least once before the time slot t, that is:

$$f^t_w = P(X^t_w \geq 1) = 1 - P(X^t_w = 0) = 1 - (1 - p_w)^t \quad (1)$$

$(X^t_w)_{\{t \in I, w \in W\}}$ are independent random variables representing the numbers of intruders direct exploitations. Each random variable yields to an exploit with probability p_w at each time $t \in I$, following a binomial distribution with parameters t and p_w. The function (1) is increasing over time. This choice is motivated by the fact that the more time passes the easier it is for an intruder to exploit a vulnerability.

Now, the impact I_w of a node $w = (v, a) \in W$ is defined as the level of damage generated by exploiting v on a. In this paper, we consider the impact to remain constant over time. CVSS scoring method [10] is used to give an estimate of the impact I_w, and the exploitation p_w.

The application Δ is defined such that for each node $n = (v, a) \in W$, gives its associated asset $a \in \Lambda$.

$$\Delta : \underset{w=(a,v)}{W} \underset{\mapsto}{\rightarrow} \underset{a}{\Lambda}$$

An arc from $w_1 = (a_1, v_1)$ to $w_2 = (a_2, v_2)$ exists if the exploitation of v_1 on a_1 makes possible the exploitation of v_2 on a_2. A direct exploitation of a vulnerability v on an asset a from an access point $u \in U$ is represented by an arc from u to $w = (a, v)$. An indirect exploitation corresponds to an $u - w$ path in G_t.

Let $t \in I$. Each arc $(n_1, n_2) \in A_t$ is evaluated by the accessibility function denoted by $g^t_{(n_1,n_2)}$. The function g between $n_1, n_2 \in V$ at time t is defined as a fraction of the connection time between the assets $\Delta(n_1)$ and $\Delta(n_2)$ during the time period from t to $t + 1$. This metric indicates how much time the assets are accessible between each other, in a given period of time,

The arcs of the RAGs could be perturbed by the access between the assets, which modifies the value of the accessibility function. So, the arcs of A_t are partitioned into two subsets A_f and A^t_u. On the one hand, the subset A_f is a fixed subset of arcs. An example of arcs that belong to A_f are those linking the pairs of nodes associated with the same asset. Indeed, since an asset is always accessible from itself over the time, we have $\forall t \in I \; g^t_{(n_1,n_2)} = 1$ if $\Delta(n_1) = \Delta(n_2)$. On the other hand, A^t_u represents the uncertain arcs that might exist or not at each time slot in I. In particular, if at a time t, we have $g^t_{(n_1,n_2)} = 0$, for a couple of nodes $n_1, n_2 \in V$, the arc (n_1, n_2) is deleted.

3. RISK EVALUATION BASED ON WORST CASE SCENARIO

Our risk evaluation approach is based on a worst case scenario: the intruders are propagating according to the most likely path. This is the path that has a maximum risk propagation.

More formally, let $\pi_{u,w}$ define the set of paths from an access point $u \in U$ to an asset-vulnerability node $w \in W$. Let $\pi = (n_1, \dots, n_k)$ be a path of length k in $\pi_{u,w}$, where $n_1 = u$ and $n_k = w$. We define *the propagated potentiality* P in π as follows.

$$P^{\pi,t}_{u,w} = \prod_{i=1}^{k-1} f^t_{n_{i+1}} \times g^t_{(n_i,n_{i+1})} \quad (2)$$

The problem of finding the most likely path is then formulated as:

$$P^t_{u,w} = \underset{\pi \in \pi_{u,w}}{max} \{P^{\pi,t}_{u,w}\} \quad (3)$$

Let us show how to compute the value of the most likely path. The problem (3) can be formulated as a shortest path problem [8]. Let $t \in I$, and let $(n_1, n_2) \in A_t$. We define *the Propagation difficulty function* on the arc (n_1, n_2) at time t as:

$$H^t_{(n_1,n_2)} = -log(f^t_{n_2} \times g^t_{(n_1,n_2)}) \quad (4)$$

The problem of finding the propagated potentiality $P^t_{u,w}$ is then equivalent to

$$P^t_{w,u} = \underset{\pi \in \pi_{u,n}}{min} \{\sum_{i=1}^{k-1} H^t_{(n_i,n_{i+1})}\} \quad (5)$$

To compute the value of the most likely path at each time slot $t \in I$, we simply label the arcs of G_t by the difficulty propagation function H. Then, by running a shortest path algorithm on G_t, the length of the shortest path, denoted by

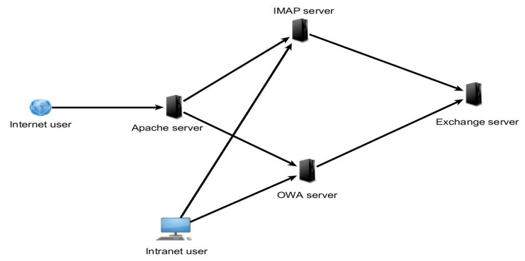


Figure 1: Email Enterprise Use Case

sp, has the same magnitude as the value of the most likely path. In fact, $P^t_{u,w} = \frac{1}{exp(sp)}$.

Now we define *the propagated risk*, *the node risk*, and *the global risk*. The risk of having a successful attack on $w = (a, v)$ is a potential exploitation of the vulnerability v. This exploitation has an impact on the affected assets. The propagated risk is:

$$R^t_{u,w} = P^t_{u,w} I_w \tag{6}$$

the node risk is defined as follows.

$$R^t_w = \sum_{u \in U} R^t_{u,w} \tag{7}$$

And the global risk is:

$$R^t = \sum_{w \in W} R^t_w \tag{8}$$

4. ENTERPRISE EMAIL CASE STUDY

We illustrate the use of our framework through the simple example of an enterprise email service. As in Figure 1, we consider two kind of users which are the internet user and the intranet user representing respectively an external and an insider intruder. Both users aim at attacking the exchange server. While the insider attacker does this through accessing either the IMAP server (Internet Message Access Protocol server) or the OWA server (Outlook Web Access sever), the external attacker should first pass through an Apache server. The assets of the system are named using the standard Common Platform Enumeration (CPE).

Table 1: Topology and Vulnerability Data Basis Mapping

Assets	Vul.	Name	p_w	I_w
a_1	v_1	CVE-2014-0098	0.5	2.9
	v_2	CVE-2013-6438	0.5	2.9
a_2	v_3	CVE-2011-3208	0.5	6.4
a_3	v_4	CVE-2013-3870	0.43	10
a_4	v_5	CVE-2014-6319	0.5	2.9

4.1 Risk Assessment Graphs

We set an example of the construction and investigation of RAGs through email enterprise service of Figure 1. We take a time interval $I = \{1, \ldots, 4\}$. The apache server, the IMAP server, the OWA server, and the exchange server are respectively denoted by a_1, a_2, a_3, and a_4. For the sake of readability, we restrict ourselves to five vulnerabilities, one for each asset except a_1, which has two vulnerabilities. Table 1 contains a description of the vulnerabilities and their associated assets at the initial state of the system ($t = 1$). The exploitation and the impact of vulnerabilities are derived from the National Vulnerability Data Base (NVD)[5].

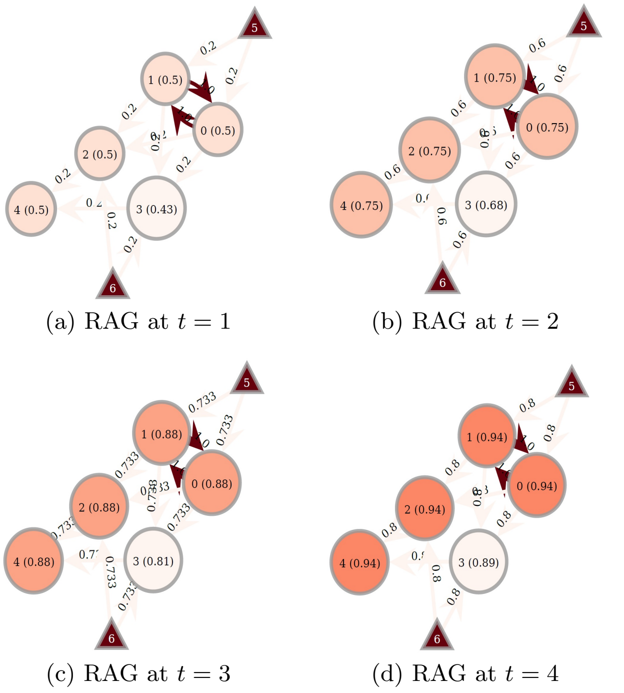
The potentiality function (1) is used to label the nodes of the RAGs. In this example, the accessibility between the nodes of the RAGs are increasing functions of time simulated as follows.

$$g^t_{(n_1,n_2)} = a_{(n_1,n_2)} + (1 - a_{(n_1,n_2)})\frac{\beta(t-1)}{t} \tag{9}$$

The term $a_{(n_1,n_2)}$ is the accessibility between nodes n_1 and n_2 at the initial system state t=1, and the parameter β controls how fast the accessibility tends to 1. We set $\beta = 1$, and $a_{(n_1,n_2)} = 0.2$, $\forall (n_1, n_2) \in A_t$, $\forall t \in I$.

The visualization of the RAGs over the time is illustrated in Figures 2(a), 2(b), 2(c), and 2(d). For each asset, we construct as many nodes in the RAG as its associated vulnerabilities. We obtain 5 asset-vulnerability nodes illustrated as circles, and refereed by $0, 1, 2, 3$ and 4, corresponding respectively to $(a_1, v_1), (a_1, v_2), (a_2, v_3)$, (a_3, v_4), and (a_4, v_5). The nodes 5 and 6 correspond to the internet and the intranet users, which play the role of system access points and illustrated as triangles.

In Figure 2 each asset-vulnerability node is labelled with the potentiality. The higher the potentiality of a node is, the darker is its color. An arc (n_1, n_2) is drawn if $g^t_{(n_1,n_2)} \neq 0$. The nodes 0 and 1 correspond to the same asset, and so are connected by a bidirectional edge. The arcs are labelled by the accessibility function (9). Since the potentiality and accessibility functions are increasing functions over time, we consequently observe that the colors of the nodes and the arcs become darker as time passes from Figure 2(a) to Figure 2(d).



(a) RAG at $t = 1$

(b) RAG at $t = 2$

(c) RAG at $t = 3$

(d) RAG at $t = 4$

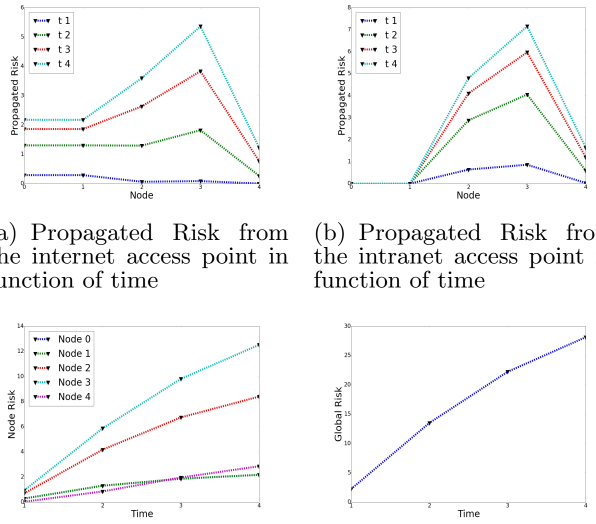
Figure 2: Visualization of the RAGs

4.2 Risk Evaluation

In Figures 3(a), and 3(b), we study over the time interval $I = \{1, \ldots, 4\}$, the propagated risk variation of the internet access point (the node 5 in Figure 2) and the intranet access point (the node 6 in Figure 2). A vertical lecture of these Figures shows, as expected, that the propagated risk from the internet and the intranet users to each asset-vulnerability node, increases over time.

For each time slot in I, the propagated risk of the internet access point (Figure 3(a)) decreases from one node to another in this order: $3, 2, 1, 0$ to 4. We also observe that for each time slot in I, the intranet propagated risk to the nodes $0, 1$ is zero. This is due to the absence of a topological access from the access point 5 to the nodes $\{0, 1\}$ (no path in the RAGs).

As seen in Figure 3(c), for each node, the risk is increasing in time. Furthermore, at each time slots in I, the node risk has the same value for the nodes $\{0, 1\}$, since the propagated risks to the two latter nodes are equal. The risk of the nodes $\{0, 1\}$ is bigger than the risk of the node 4 before the time slot 3. Nevertheless, the situation is reversed when $t = 4$. This is explained by the fact that the propagated risk from the intranet user to the nodes $\{0, 1\}$ is null, while the one from the same user to the node 4 continue to increase from a time slot to another. Finally, we plot the global risk variation in Figure 3(d), which is increasing over time.



(a) Propagated Risk from the internet access point in function of time

(b) Propagated Risk from the intranet access point in function of time

(c) Node Risk in function of time

(d) Global Risk in function of time

Figure 3: Risk Evaluation

5. CONCLUSION AND ONGOING WORK

In this paper, we propose a new risk assessment framework to supervise the status of complex ICT systems. We introduce the concept of Risk Assessment Graphs (RAGs) which captures the topological information, including the assets of the system and the accessibilities between them, vulnerabilities associated with each asset, as well as the way these elements vary over time. We also provide a worst-case risk evaluation approach based on the propagation of the intruder threats through the most likely path. We defined three security metrics namely the propagated risk, the node risk, and the global risk. We use the simplified example of an enterprise email case study to demonstrate the use of our approach throughout this paper.

This framework allows to identify in which time slots the system is not secured. An alert could then be sent to start control actions. The latter may consist in the deployment of countermeasures on an asset in order to prevent the propagation of intruder threats. However, the deployment of a countermeasure may have a prohibitive cost when compared to the encountered risks. In our ongoing works we investigate the use of combinatorial optimization [13] to deduce robust control actions from the RAGs.

6. REFERENCES

[1] Common platform enumeration, cpe.

[2] Ebios, central directorate for information systems security, version 2010 website.

[3] Mehari, method harmonized risk analysis.

[4] Nist, national institute of science and technology.

[5] Nvd, national vulnerability database.

[6] P. Ammann, D. Wijesekera, and S. Kaushik. Scalable, graph-based network vulnerability analysis. In *Proceedings of the 9th ACM Conference on Computer and Communications Security*, pages 217–224. ACM, 2002.

[7] J. A. Bondy and U. S. R. Murty. *Graph theory with applications*, volume 290. Macmillan London, 1976.

[8] R. W. Floyd. Algorithm 97: shortest path. *Communications of the ACM*, 5(6):345, 1962.

[9] H. Huang, S. Zhang, X. Ou, A. Prakash, and K. Sakallah. Distilling critical attack graph surface iteratively through minimum-cost sat solving. In *Proceedings of the 27th Annual Computer Security Applications Conference*, pages 31–40. ACM, 2011.

[10] P. Mell, K. Scarfone, and S. Romanosky. Common vulnerability scoring system. *Security & Privacy, IEEE*, 4(6):85–89, 2006.

[11] X. Ou, W. F. Boyer, and M. A. McQueen. A scalable approach to attack graph generation. In *Proceedings of the 13th ACM conference on Computer and communications security*, pages 336–345. ACM, 2006.

[12] C. Phillips and L. P. Swiler. A graph-based system for network-vulnerability analysis. In *Proceedings of the 1998 workshop on New security paradigms*, pages 71–79. ACM, 1998.

[13] A. Schrijver. *Combinatorial optimization: polyhedra and efficiency*, volume 24. Springer Science & Business Media, 2002.

[14] V. Viduto, W. Huang, and C. Maple. Toward optimal multi-objective models of network security: Survey. In *Automation and Computing (ICAC), 2011 17th International Conference on*, pages 6–11. IEEE, 2011.