A New Bayesian Approach to Exploring Damaged Assets by Monitoring Mission Failures Caused by Undetected Attack

Shinwoo Shim[1,2] and Ji Won Yoon[1](✉)

[1] Graduate School of Information Security, Korea University,
Seoul, Republic of Korea
{shimshinwoo, jiwon_yoon}@korea.ac.kr
[2] Cyber Warfare R&D Lab, LIG Nex1 Co., Seongnam, Republic of Korea

Abstract. Modern military systems operated with a complex of computers and software may have mission failure which is caused by undetected attacks. In such situations, it is important to find out which assets are damaged. After identifying damaged assets, we need to immediately examine the damaged assets to defend against the attacks. However, it is not straightforward to explore the damaged assets because there are the complicated relationships among assets, tasks and missions. In this paper, we propose an effective methodology to infer the damaged assets given observed mission impacts in a Bayesian framework. We used Bayesian networks to model assets, tasks, missions and to set the relationships among them. Our approach visually infers and identifies the damaged assets with the probability. We show that proposed Bayesian framework is practical and useful with the use case experiment.

Keywords: Mission Impact Assessment · Bayesian network · Cyber warfare

1 Introduction

As more information technology appliances are used, it is more difficult to build effective situation awareness system which detects significant but unpredictable operational risks. This situation makes the system operators struggle to find out how the asset damages can affect missions. Therefore, defining the relationships among assets, task, missions and assessing asset damage, impact propagation are essential for military systems. There have been several researches on Battle Damage Assessment (BDA) and Mission Impact Assessment (MIA) to find out the current ability to perform missions under the asset damages.

Previous researches focused on the damage propagations from assets to tasks and missions. In such situations, a monitoring systems or sensors should detect the damages on the assets first. However, there can be attacks that cannot be detected by the monitoring systems or sensors. For example, if an enemy exploits a zero-day vulnerability and succeeds in the attack, the mission can fail while the damage on an asset is still not detected [1]. Such attacks can lead to the failure of missions without observing any damage on assets. Even though the enemy does not use a zero-day vulnerability, it

B. B. Kang and J. Jang (Eds.): WISA 2018, LNCS 11402, pp. 185–196, 2019.
https://doi.org/10.1007/978-3-030-17982-3_15

is hard to apply security patches to all the assets in time if there are hundreds of assets to manage.

If mission failure is caused by undetected asset damages, we should find out which assets were damaged. If there are a few assets for a mission, we can check all the assets one by one. However, if there are hundreds of assets involving a mission, it is impossible to check all the assets in a limited time. In this point of view, it is valuable to find undetected damaged assets by observing a mission failure. In this paper, we infer the damaged assets from an observation of the mission impact using a Bayesian network.

The main contribution of this paper can be summarized as follows: We introduce a new way in analyzing the relationships among assets, tasks and missions. Previous researches tried to assess the mission impact when asset damages are detected. Instead of assessing asset damage propagation, we focused on making damaged assets inference from observing mission impacts when there is mission failure caused by undetected attacks. To infer the damaged assets, we propose a Bayesian framework which can be constructed practically using Logical AND and Noisy-OR relationships. We can therefore infer the damaged assets by getting the probabilities of damages for each asset. We show that this methodology is practical and useful with use case experiments.

The remainder of this paper is structured as follows. In Sect. 2, we will review the related researches that have been done on the Mission Impact Assessment and Bayesian networks. In Sect. 3, we define the terms used throughout this paper. In Sect. 4, we describe the method for building a Bayesian network, which consists of assets, tasks, mission and their relationships. We show how we make damaged assets inference from an observation of a mission impact by constructing a Bayesian network. In Sect. 5, we describe the implementation of the Bayesian network and result of the experiments.

2 Related Work

Jakobson [2] proposed a conceptual framework and a method for assessing impact that cyber attack might have to cyber assets, services, and missions. The framework builds the model of a mission, service and assets, and impact dependency graph. It presents an algorithmic base how to calculate impacts that cyber attack cause, how the direct impacts propagate through the service, and mission dependencies and affect the operational capacity of missions.

Sun et al. [3] introduced System Object Dependency Graph (SODG) to capture the intrusion propagation process at low operating system level. On the top of the SODG, a mission-task-asset (MTA) map can be established to associate the system objects with tasks and missions. A Bayesian network for MTA can be constructed and it can be used to find missions being tainted and to assess quantitative mission impact.

Motzek et al. [4] proposed a mathematical mission impact assessment, based on a probabilistic approach using mission dependency models and resource dependency model. In mission dependency models, a Bayesian network is used for building a probabilistic dependency models and for assessing mission impact.

Previous works have focused on studying on how the asset damages propagate to missions. Previous works assume that the attacks are detected and then the mission impacts are calculated according to the damages. However, in this paper we assume the situation that we are not able to detect the attacks when missions fail. The goal of our study is to find the damaged assets when we observe mission failures caused by undetected attacks.

3 Terminology

Before we state the methodology, we define the word used throughout the paper.

- Mission - A set of tasks that fulfills a purpose or duty
- Task - A piece of work done as part of a mission
- Asset - Hardware or software that supports one or more tasks
- Impact - An quantitative assessment of how much a mission is affected by a given activity or situation [5]
- Damage - A quantitative assessment corresponding to the state(s) a given asset is in with respect to its ability to perform a given role [5]
- Vulnerability - A specific weakness in the protections or defenses surrounding assets

For example, “surface to air defense” for an area can be a mission and “detecting air tracks”, “intercepting enemy fighter” can be tasks which support the mission. Equipment which comprises radar system like “transmitter”, “receiver” and “signal processor” can be assets that support the task “detecting air tracks”. An asset can support one or more tasks and a task can support one or more missions.

“Impact” and “Damage” seem to be similar terms. However, they are not the same. Impact is generally the result of some damage [6]. In this paper, we use “damage” for negative influence on assets and “Impact” for negative influence on tasks and missions.

4 Damaged Assets Inference Using Bayesian Networks

Bayesian networks are probabilistic models based on directed acyclic graphs and have capability for bidirectional inferences which can model the top-down (semantic) and bottom-up (perceptual) combination of evidence [7]. A key feature of Bayesian networks is the ability to locally interpret individual parameters, i.e. to locally interpret individual probabilities of conditional probability distributions [4]. This feature provides a direct understandability to all conditional probabilities, i.e. we can make an inference for undetected damaged assets from observing impacted missions using Bayesian networks.

We need to construct a Bayesian network to find out the damaged assets which are undetected when a mission fails. Figure 1 shows a Bayesian network for assets, tasks, missions and their relationships. In this example, we constructed a small network that has four assets, three tasks, and two missions for easy explanation.

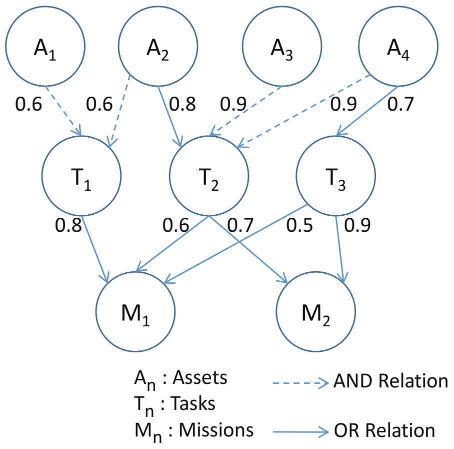

Fig. 1. A Bayesian network for assets, tasks, missions

When applying this methodology on the real military systems, experts who design or operate the military system are supposed to identify assets, tasks and missions and to construct the Bayesian network. The construction of a Bayesian network should be done by the experts who thoroughly understand the system to obtain the correct result.

Figure 1 is an example of a simple Bayesian network that describes assets, tasks, missions and the damage propagation. Figure 1 has is a little different from a typical Bayesian network. It has the probabilities on the edges instead of having conditional probability tables and it has the edges with AND or OR relations that a typical Bayesian network does not have. We will describe the notation of the Bayesian network and its application in the following sections.

4.1 Nodes and Edges

In Fig. 1, circles are nodes that denote assets, tasks and missions. For example, A_1 means the first asset, T_2 means the second task and M_1 means the first mission. We denoted assets, tasks and missions by number for simple notation. However, stating explanatory phrases like “surface to air defense” for nodes would be intuitive when building a Bayesian network.

Directed lines are edges that denote the probabilities that the child nodes are impacted when a parent node is damaged or impacted. For example, if A_2 is damaged, the damage can propagate to T_2 for the probability of 0.8. If T_3 is impacted, the impact can propagate to M_1 for the probability of 0.5 and to M_2 for the probability of 0.9.

Our purpose is not to figure out how much the assets are damaged with exact figures but to find out which assets are more probably damaged. Therefore, we define the state of the node to be binary, i.e. the states of assets can be damaged (True) or undamaged (False). The states of tasks and missions can be impacted (True) or not impacted (False).

4.2 Edge Relation

Military assets can be targets of enemies and tend to be attacked by them. Therefore, many military systems are constructed with redundancy. For example, a radar system can have two transmitters for high availability. In such a radar system, it can fulfill the task even though one transmitter is out of order and it can be impacted when both transmitters are out of order.

We depict redundancy relations with dashed edges in Bayesian networks. Parent nodes with dashed edges are in “AND Relation” which means that all the parent nodes should be damaged or impacted to impact the child node. For example, in Fig. 1, if A_3 is damaged and A_4, A_2 are not damaged, then T_2 is not impacted. Such concept of relations was also described in [2]. There can be two or more redundant assets in real situation, so two or more nodes can be in the same AND relation.

In addition to the concept in [2], we revised the AND relation to have the concept of groups. For example, in radar system stated above, a radar system can have two transmitters, and two receivers for high availability. In this situation, two transmitters can be two assets in a same AND group and two receivers can be two assets in another AND group. In such a radar system, it can fulfill the task even though one transmitter and one transmitter are out of order.

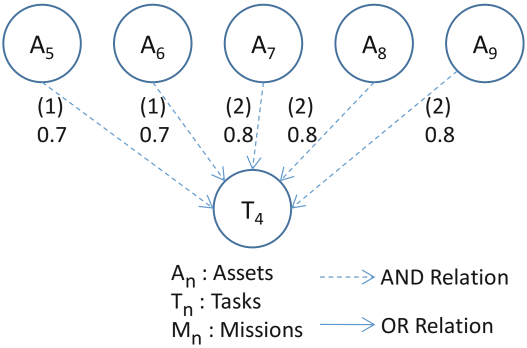

Fig. 2. Example of AND relation groups

An example for AND group is shown in Fig. 2 and it depicts a sub-graph of a Bayesian network. In Fig. 2, A_5, A_6 perform the same functions, and A_7, A_8, A_9 perform other same functions. Then $\{A_5, A_6\}$ becomes an AND group and $\{A_7, A_8, A_9\}$ becomes another AND group. Task T4 will not be impacted if at least one asset in each group is not damaged.

Relation other than “AND Relation” is “OR Relation”, which means that only one damage or impact of parent nodes can affect the child node. We explain how the probabilities are calculated for AND relation and OR relation in Sect. 4.3.

4.3 Conditional Probability Tables (CPTs)

We should build conditional probability tables (CPTs) to make inference from a Bayesian network. CPTs state the probabilities of each node as the state of parent nodes changes. For example, Table 1 expresses the CPT for T_2 in Fig. 1. As we can see in Table 1, if a node has n parents, there are 2^n entries for the CPT of the node.

Table 1. Conditional probability table for T_2

A_2	A_3	A_4	T_2	
			T	*F*
F	F	F	0	1
F	F	T	0	1
F	T	F	0	1
F	T	T	0.81	0.19
T	F	F	0.8	0.2
T	F	T	0.8	0.2
T	T	F	0.8	0.2
T	T	T	0.962	0.038

The entries of the CPT increase exponentially as the parent nodes increase. For example, if a task is affected by 10 assets, there exists $2^{10} = 1024$ entries in a CPT. We stated that the construction of a Bayesian network should be done by experts. In this case, experts should enter 1024 entries for one node, and it is impossible for the human to enter all the entry. To resolve this difficulty, we adopt the concept of Noisy-OR [9, 10] and logical AND [8]. We can reduce the entities that the experts should fill out using logical AND and Noisy-OR methodology. The probabilities in each entry can be calculated by the formulas stated in the following sections. As a result, the experts only have to grade the probabilities of each edge instead of filling out all the entities in conditional probability tables.

Probability for AND Relation. Logical AND is applied for “AND Relation” and it means that damage can propagate only if all the parent nodes which have the “AND Relation” of same groups are damaged or impacted. The formula for logical AND can be expressed as:

$$p(X_i) = \begin{cases} 0, & \text{if at least one } X_j \text{ is False} \\ \prod_{j:X_j} p(e_i), & \text{otherwise} \end{cases} \tag{1}$$

$p(X_i)$ is the probability distribution of child node X_i. X_j is the parent node whose value is True. And e_j is the edge between child node X_i and parent node X_j and $p(e_j)$ is the probability of e_j.

Probability for OR Relation. Noisy-OR is applied for "OR Relation" and it means that damage or impact of only one parent node can affect the child node. The formula for Noisy-OR can be expressed as:

$$p(X_i) = \begin{cases} 0, & \text{if all } X_j \text{ is False} \\ 1 - \prod_{j:X_j} (1 - p(e_i)), & \text{otherwise} \end{cases} \tag{2}$$

The notation is same as logical AND.

Generalized Probability. In our Bayesian network, 'AND Relation' and 'OR Relation' can exist together and there can be several groups of 'AND Relation' edges for one child node. Therefore, the formula can be generalized as:

$$p(X_i) = \begin{cases} 0, & \text{if all } Y_k \text{ is False} \\ 1 - \prod_{k:Y_k} (1 - Y_k), & \text{otherwise} \end{cases} \tag{3}$$

where,

$$Y_k = \begin{cases} p(e_k), & \text{if } e_k \text{ is an OR relation edge} \\ \prod_{n:X_n} p(e_n), & \text{if } X_n \text{ is in AND group with n nodes} \end{cases}$$

Y_k is the parent node of X_i and other notation is same as Logical AND and Noisy-OR. The example of a CPT for logical AND and noisy-OR is depicted in Table 1. It shows the CPT for task T_2 in Fig. 1. A_3 and A_4 are in AND relation for T_2 and the edge probabilities are 0.9 each. One edge from A_2 to T_2 is in OR relation and has the probability of 0.8. When A_2, A_3, A_4 are all damaged (True), the possibility can be calculated using Eq. (3) and the result is 1 − (1 − 0.9 * 0.9) * (1 − 0.8) = 0.962.

Risk of Assets. Risk of assets should be identified prior to the assessment. Risk of assets means how the assets are vulnerable or how the assets can be easily attacked by enemies. Risk of assets can be expressed as the probability of the assets to be damaged, i.e. $P(A_n)$.

The 'health' of the assets can be measured by security posture metric (SPM) [11]. SPM can be calculated by Common Vulnerability Scoring System (CVSS) [12] and SPM is a value between 0 and 1, where 1 represents a 'safe' asset. The formulation of SPM is shown in Eq. (4).

$$SPM(A_n) = \frac{1}{\sum_j \frac{1}{\left(1 - \frac{CVSS_{n,j}}{11}\right)}} \tag{4}$$

$CVSS_{n,j}$ is the score of vulnerability A_n has. Scores range from 0 to 10, with 10 being the most severe. By using SPM we can calculate the probability of the asset risk as shown in Eq. (5).

$$\mathrm{P}(\mathrm{A_n}) = \mathrm{SPM}(\mathrm{A_n}) \tag{5}$$

This probability of an asset means how easily the enemy can damage or compromise the asset. In this paper, we used CVSS score for the quantitative measure and other methods can be used for calculating risk probabilities of assets.

4.4 Finding Damaged Assets

If we have the risk probabilities of each asset i.e. $\mathrm{P}(\mathrm{A_n})$ and the CPTs for all tasks and missions, we can make an inference of damaged assets from observation of mission impacts. The probabilities of an asset damage given a mission failure can be calculated as shown in Eq. (6).

$$\mathrm{P_i}(\mathrm{A_n}) = \mathrm{P}(\mathrm{A_n}|\mathrm{M_i} = \mathrm{T}) \tag{6}$$

$\mathrm{P_i}(\mathrm{A_n})$ is the conditional probability of asset $\mathrm{A_n}$ given the condition that mission $\mathrm{M_i}$ is impacted. For example in Fig. 1, if $\mathrm{M_1}$ is impacted, the probability that $\mathrm{A_2}$ can have damage is $\mathrm{P}(\mathrm{A_2} \mid \mathrm{M_1} = \mathrm{T})$ and it can be calculated by summing out other nodes.

We can get the damage probabilities of all the assets and we can rank them. As a result, we can prioritize assets we should examine first when we observe a mission impact without asset damage detection.

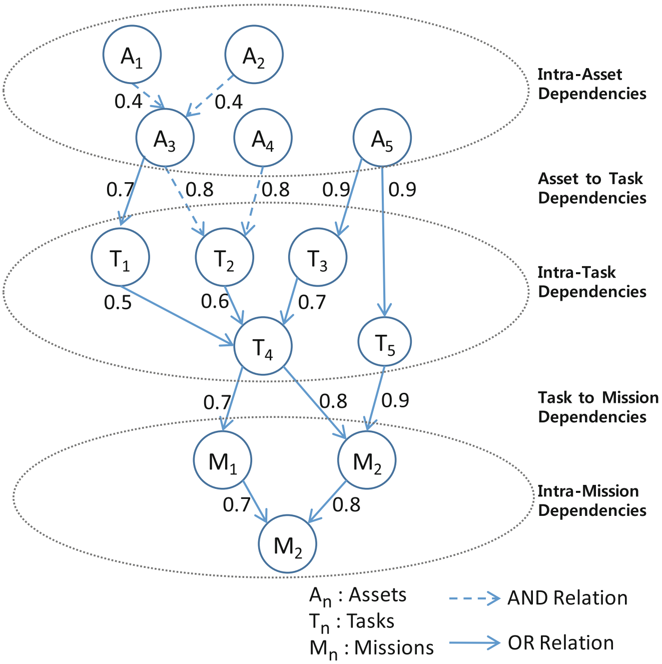

Fig. 3. An example of a Bayesian network that has intra-dependency relationships

4.5 Extension of Bayesian Networks

So far, we explained the Bayesian network which has inter-dependency, i.e. the dependency exists between asset nodes and task nodes, also between task nodes and mission nodes.

In our Bayesian framework, we can extend the network to have the concept of intra-dependency introduced in [2]. Intra-dependency means that there can be relationships among assets and also among tasks and missions.

The example of a Bayesian network that has intra-dependency is shown in Fig. 3. In this example, the graph is still directed acyclic, so our Bayesian framework can work by the methodology we described above. However, if there are nodes which have mutual dependency, then the graph becomes cyclic and the Bayesian framework would not work.

5 Experiments

In this section, we apply our methodology on a real world use case. The code implementation was made on MATLAB R2016b using Bayes Net Toolbox [13].

We present a simplified and abstract weapon system using a Bayesian Network described in Fig. 4. There are two missions, which are surface-to-air defense and the air-to-air defense and there exists two weapon systems, which are a surface-to-air defense system and an air-to-air defense system for each mission. Each mission comprises three tasks, which are detecting air targets, intercepting air targets and sharing information. Detecting air targets is a role of radar system and the assets can be signal processors and transmitters/receivers. To intercept air targets, missile launchers are needed for the assets. Sharing information is needed for exact situation awareness and communication equipment like tactical data link equipment is needed as assets. The explanation for each node is shown in Table 2.

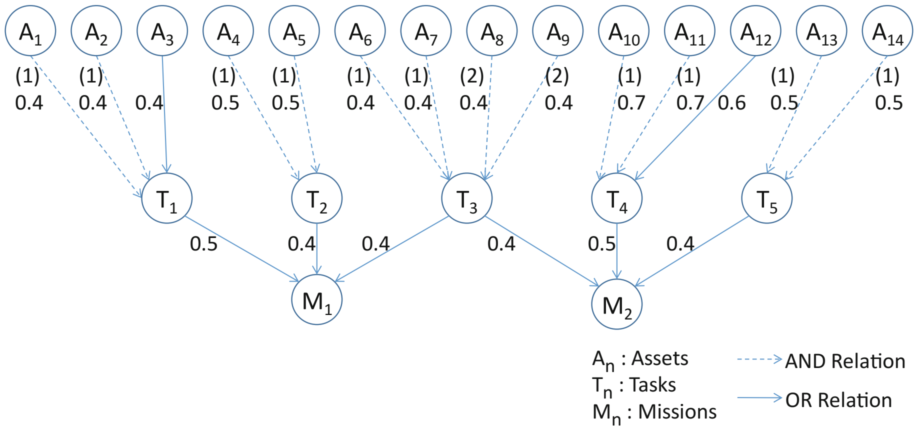

Fig. 4. A Bayesian network for air defense missions

As shown in Fig. 4, some assets have redundancy and the others do not have redundancy. For the simplification, we assume that the assets have the same risk probabilities for assets and impact propagation probabilities have similar values as shown in Fig. 4. We supposed that all the risk probability of asset is 0.3.

Table 2. Description for the nodes in the Bayesian network for air defense missions

Label	Description	Weapon system	Category
A_1	Transmitter/Receiver of radar	Surface-to-Air defense	Asset
A_2			
A_3	Signal processor of radar		
A_4	Missile launcher		
A_5			
A_6	Tactical data link equipment		
A_7			
A_8	Tactical data link equipment	Air-to-Air defense	
A_9			
A_{10}	Transmitter/Receiver of radar		
A_{11}			
A_{12}	Signal processor of radar		
A_{13}	Missile launcher		
A_{14}			
T_1	Detecting air targets	Surface-to-Air defense	Task
T_2	Intercepting air targets		
T_3	Sharing information	Surface-to-Air defense Air-to-Air defense	
T_4	Detecting air targets	Air-to-Air defense	
T_5	Intercepting air targets	Air-to-Air defense	
M_1	Surface-to-Air defense	Surface-to-Air defense	Mission
M_2	Air-to-Air defense	Air-to-Air defense	

After building a Bayesian network, we assumed that surface-to-air defense mission failed and then calculated the possibilities of the asset damages.

If we do not have the Bayesian framework and all the information we have is just predefined fixed possibility of asset vulnerability, we should examine all the assets one by one in a brute-force way to find out the damaged asset because all the risk probability for the asset is the same. However, we can estimate the probabilities of the asset damages based on the condition that the mission is failed. The probabilities of the asset damages after the failure of the surface-to-air defense mission are shown in Fig. 5.

The most probabilities of assets damages for the surface-to-air defense weapon system are raised by 8%–26% compared to the original probability. However, the probability of the signal processor of radar (A_3) has been raised by more than 170%. If the attacker intends to attack the transmitter/receiver, then the attacker should attack two transmitters/receivers at the same time because they have redundancy relationships.

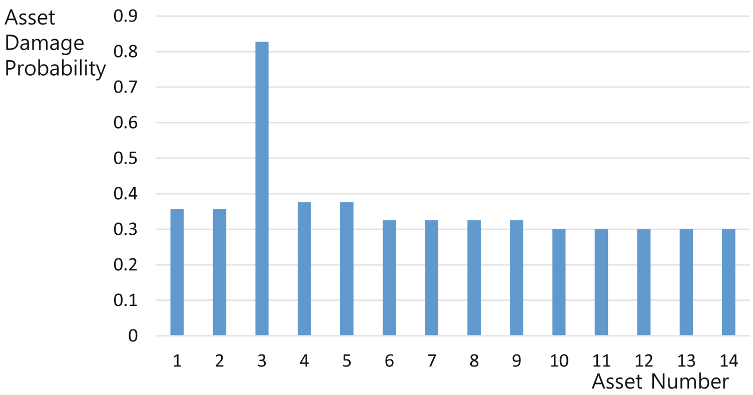

Fig. 5. Probabilities of assets damages based on the condition that the mission is failed

On the other hand, if the attacker intends to attack the signal processor, the attacker has to attack only one asset because the signal processor does not have a redundancy. Therefore, the Bayesian framework analyzes that it is easier for the attacker to attack the signal processor and raises its damage probability. Using the result of the analysis we can prioritize the assets to examine first.

In this use case, it can be intuitive to find out the most probable asset that an attacker would aim because the network is small and the relationships are not complicated. However, if the network becomes larger and the relationships get more complicated, it would be impossible to find out the most probable damaged asset by human intuition. In such cases, our Bayesian framework can prioritize the assets to be examined first.

6 Conclusion

In this paper, we have proposed a Bayesian network model for assets, tasks and missions and their relationships to infer damaged assets from observing an impacted mission. This methodology enables prioritizing the assets to be examined first when we fail to detect the asset damages.

We used an intuitive and mathematical method in modeling Bayesian networks and calculating the probabilities. Building Bayesian networks and setting conditional probability tables are intuitive and feasible for experts who are responsible for managing them. The experts define the assets, tasks, missions and set the relationships among them by AND relation or OR relation edges and then set probabilities of the edges that mean the impact probabilities of child nodes when the parent nodes are damaged or impacted. Instead of entering all the entities of conditional probability tables, the conditional probability tables are set automatically using the formula based on the edge values and the relationships among them. We showed that the inference is feasible and practical by the use case experiments.

We are continuing our research for the Bayesian framework which has mutual dependency and for other complementary researches relating issues on finding damaged assets.

References

1. Bilge, L., Dumitras, T.: Before we knew it: an empirical study of zero-day attacks in the real world. In: Proceedings of the 2012 ACM Conference on Computer and Communications Security, pp. 833–844 (2012)
2. Jakobson, G.: Mission cyber security situation assessment using impact dependency graphs. In: Proceedings of the 14th International Conference on Information Fusion, Chicago, IL (2011)
3. Sun, X., Singhal, A., Liu, P.: Who touched my mission: towards probabilistic mission impact assessment. In: Proceedings of the 2015 Workshop on Automated Decision Making for Active Cyber Defense, SafeConfig 2015, pp. 21–26, New York, NY, USA. ACM (2015)
4. Motzek, A., Möller, R.: Context- and bias-free probabilistic mission impact assessment. Comput. Secur. **65**(2017), 166–186 (2017)
5. Holspopple, J., Yang, S.J.: Handling temporal and functional changes for mission impact assessment. In: Proceedings International Multidisciplinary Conference on Cognitive Methods in Situational Awareness and Decision Support (CogSIMA), San Diego (2013)
6. Fortson, L.W.: Towards the development of a defensive cyber damage and mission impact methodology. AFIT Masters thesis, March 2007
7. Pearl, J., Russell, S.: Bayesian networks. In: The Handbook of Brain Theory and Neural Networks, 2nd edn, pp. 157–160. MIT Press (2003)
8. Munoz-Gonzalez, L., Sgandurra, D., Barrere, M., Lupu, E.C.: Exact inference techniques for the analysis of Bayesian attack graphs. IEEE Trans. Dependable Secur. Comput. **16**, 231–244 (2017)
9. Xie, P., Li, J.H., Ou, X., Liu, P., Levy, R.: Using Bayesian networks for cyber security analysis. In: 2010 IEEE/IFIP International Conference on Dependable Systems & Networks (DSN), Chicago, IL, pp. 211–220 (2010)
10. Henrion, M.: Practical issues in constructing a Bayes' belief network. In: Proceedings of Third Workshop on Uncertainty in AI, AAAI, Seattle, Washington, pp. 132–139, July 1987
11. Nakhla, N., Perrett, K., McKenzie, C.: Automated computer network defense using ARMOUR: mission-oriented decision support and vulnerability mitigation. In: 2017 International Conference on Cyber Situational Awareness, Data Analytics and Assessment (Cyber SA), London, pp. 1–8 (2017)
12. FIRST Org. Inc.: Common Vulnerability Scoring System v3.0. Specification Document. https://www.first.org/cvss/
13. Murphy, K.: Bayes Net Toolbox for Matlab. https://www.cs.utah.edu/~tch/notes/matlab/bnt/docs/bnt_pre_sf.html. https://github.com/bayesnet/bnt