Towards Cyber Resiliency in the Context of Cloud Computing

Xiaoyan Sun, California State University, Sacramento, Email: xiaoyan.sun@csus.edu
Peng Liu, Pennsylvania State University, Email: pliu@ist.psu.edu
Anoop Singhal, Computer Security Division, NIST, Email:psinghal@nist.gov

Cyber resiliency is the capability of an enterprise network to continuously provide (the supported missions and business processes with) essential functions in the midst of an attack campaign. It is defined as "the ability to anticipate, withstand, recover from, and adapt to adverse conditions, stresses, attacks, or compromises on systems that include cyber resources" [1]. Conceptually speaking, the capability can be measured by whether the supported missions and business processes can succeed in spite of the various impact being caused by the attack campaign. Since the success criteria for different missions and business processes are often different, cyber resiliency is in general a *relative* notion. Enabling business process A to succeed in the midst of an attack campaign does not really mean that the enterprise network's cyber resiliency capability will also enable business process B to succeed in the midst of the same attack campaign. Although technically it is possible to define non-relative system-wide cyber resiliency criteria at the operating system level or the network service level, such cyber resiliency measurements in many cases do not directly measure the business side cyber resiliency (e.g., to which extent a task is affected by the attack campaign).

Besides being a relative notion, cyber resiliency is also a capability depending upon multiple factors. First, a business process could involve tasks which could be running on any part of the enterprise network. Accordingly, any security vulnerability, when being exploited by an attack campaign, could generate negative impact on the business process. Second, any security measure deployed on the enterprise network could help mitigate the impact on the business process. Third, this business process is usually not alone, data dependencies and control dependencies could exist between this business process and some other business processes. Accordingly, the impacts (of the attack campaign) on this business process are also related to how other business processes are impacted by the attack campaign.

Due to the fact that cyber resiliency depends upon multiple factors, although cyber resiliency is a very important security notion and competence in real-world enterprises, it is a capability that researchers have found difficult to precisely define, effort-consuming to clearly articulate, and hard to quantify or measure.

To help explain these difficulties, we argue that cyber resiliency is a delicate "balancing act" between a set of resilience indicators, including damage in terms of integrity loss, damage in terms of availability loss, situation awareness (e.g., detection), costs (e.g., resources, management costs), redundancy and diversity, dependencies, adaptation (e.g., moving target defense), quarantine, recovery, deception (e.g., honeypots), and agility (e.g., delay). Our argument is supported by the following observations:

- On one hand, the cyber resiliency problem could be theoretically solved by providing unlimited redundant computing resources and sufficient diversity. On the other hand, if we require too many redundant computing resources (to mask the impact of cyber-attack), the cost would be unacceptably high.
- How much awareness is needed is somehow related to how much redundant computing resources we have.
- Although faster and/or more accurate intrusion detection can lead to improved cyber resiliency, there is no such thing as a free lunch. For example, fine-grained taint analysis may introduce 3 times performance overhead to a web server.
- If we do not recover from the integrity loss in a timely manner, the designated functionality could be seriously hurt, leading to substantial availability loss.
- If we do not gain sufficient awareness, we won't even know whether the attack campaign has caused any unacceptable impact on a business process or not; as a result, it is hard to make any sensible decisions on taking adaptation, quarantine, and recovery actions.
- On one hand, if we do too much quarantine, the designated functionality would probably be hurt; on the other hand, if we do too little quarantine, the damage could propagate too quickly to impact the designated functionality before we quarantine it.
- If we rely too much on recovery, the availability loss could be unacceptably high; however, if we do not do any recovery, the damage spreading could soon become out of control.
- Adaptation, quarantine, and recovery could result in substantial delay in providing the designated functionality; however, if we decide to simply avoid such delays, the damage

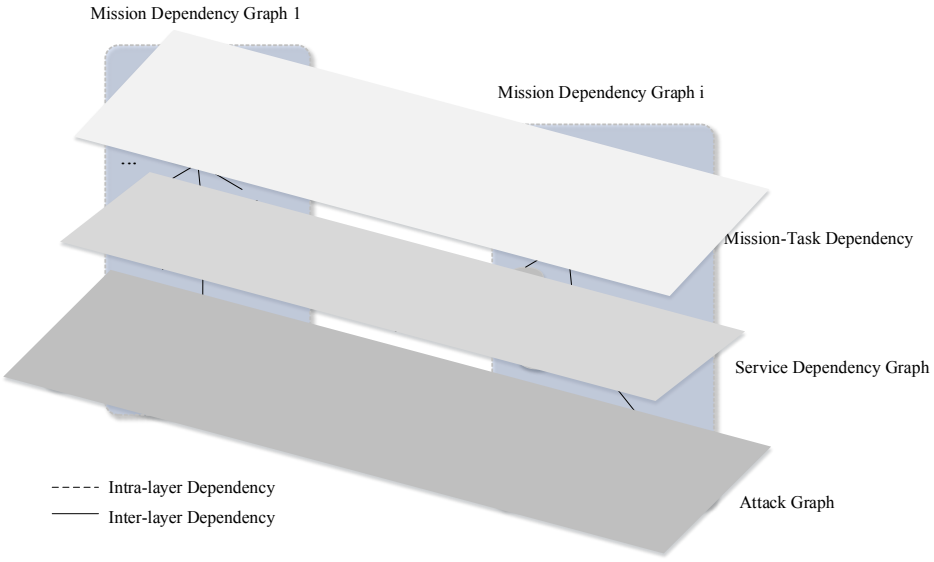

Figure 1. The Mission Dependency Graph, Service Dependency Graph, and Attack Graph (Adapted from Fig. 1 in [2]).

(and damage spreading) may quickly impact the designated functionality.

- On one hand, adaptation and deception may deter the attack campaign. This mitigates the negative effects of the aforementioned delays. On the other hand, adaptation and deception may substantially increase the system complexity and may consume a lot of resources.

An overlooked gap between existing mission impact assessment and cyber resilience techniques

Whether cyber resiliency is achieved for a business process is technically determined by mission impact assessment. How cyber resiliency is achieved is determined by the actions taken by attack-resilient systems and networks. Both mission impact assessment and attack-resilient systems have been extensively studied by the research community. However, there is still an overlooked gap between *whether* cyber resiliency is achieved (mission impact assessment) and *how* it is achieved (cyber resilience techniques).

First, from the perspective of cyber resilience, the existing cyber resilience techniques are not mission-centric. Extensive research has been performed towards attack-resilient systems and networks [3], but most if not all of the techniques do not consider the mission impact. The cyber resilience analysis is generally constrained to the level of cyber assets, and thus the resulting recommendations might not be correct and accurate if the impact towards mission is considered.

Second, from the perspective of mission impact assessment, current mission impact assessment models lack the capability of cyber resilience analysis. Therefore, the mission impact results cannot be automatically used to make mission-centric resilience recommendations on taking cyber response actions.

In cloud environment, the gap between mission impact assessment and cyber resilience becomes even more evident. The missions belonging to different enterprise networks in a public cloud should be isolated and not intervene with each other. However, due to the Virtual Machine (VM) image sharing among cloud tenants and VM co-residency on the same physical host, multi-step attacks can penetrate the boundaries between individual enterprise networks and thus impact missions of multiple enterprise networks. Hence, attacks in one enterprise network can possibly affect missions of another enterprise network on the same cloud.

Towards Bridging the Gap

With the substantial amount of prior efforts towards mission impact assessment and cyber resilience, an effective way to bridge the gap is leveraging existing models and techniques in the two areas. Since mission dependency graphs and attack graphs have been respectively developed for mission impact analysis and attack-graph-based cyber resilience, the strategy we take is to integrate mission dependency graphs and cloud-level attack graphs into a unified graphical model. Figure 1 shows the relationship among mission dependency graphs,

attack graphs and service dependency graphs. The attack graphs and service dependency graphs are horizontal graphs that cover specific abstract layers, namely the asset layer and the service layer, while the mission dependency graphs are vertical graphs that connect across multiple abstract layers.

Both mission dependency graphs and attack graphs lack some capabilities for accurate mission impact assessment.

The mission dependency graphs capture the dependency relations among entities at different abstract layers. However, the loose definition of intra- and inter- layer dependencies often leads to inaccurate or incorrect mission impact assessment. For instance, in Figure 1, assuming that mission m_1 and mission m_i transitively depends on host h_1 and host h_i respectively, the mission impact assessment through each individual mission dependency graph would lead to the conclusion: if host h_1 is attacked, mission m_1 will be impacted while m_i is intact. That's because m_i does not depend on h_1 according to the individual mission dependency graphs. However, since a multi-step attack at asset layer is potential, h_i could be compromised by taking h_1 as a stepping stone. In this case, m_i will eventually get impacted too. In addition, service dependency is also a component missing in mission impact assessment.

Attack graphs are able to show the potential attack paths by analyzing the causality relations between vulnerabilities and exploitations. Nevertheless, the traditional attack graph is limited in that it is not mission centric: it is not able to show the impact of vulnerability exploitations towards specific missions.

Mission Impact Assessment Framework

To take advantage of different graph types' capabilities and compensate for their inadequacy, we propose a mission impact assessment framework that contains two components: a new graphical model named *mission impact graph* to integrate mission dependency graph, service dependency graph, and cloud-level attack graph; and the applicable metrics on top of the graphical model to actually measure the impact. The metric quantitatively shows how much an attack or a resilience action could impact missions and can thus provide reference for making cyber resilience recommendations.

Figure 2 shows the framework of mission impact assessment. The network information is collected and fed into MulVAL [4], which is an existing attack graph generation toolkit, to generate the mission impact graph. On top of the mission impact graph, the qualitative or quantitative mission impact assessment can be performed. The assessment results could become one of the important references for security analysts to make cyber resilience recommendations. The recommended response actions, once taken, may further change the network status and thus trigger another round of mission impact assessment. Please note that making cyber resilience recommendations is a delicate balancing act that involves many factors. The mission impact assessment result is just one of the many references and does not provide decisive guidance to cyber resilience act. Other organization-specific factors, such as the limitation of human and financial resources, priority of business goals, policies, etc., should also be considered before the cyber resilience decision making.

To enable automatic generation of mission impact graphs, we extended the capability of MulVAL by crafting mission-aware interaction rules. Three set of inputs are converted to Datalog clauses and input into MulVAL, including the mission dependencies, service dependencies, and cloud-level attack related information. The reasoning engine of MulVAL then applies the interaction rules against the input Datalog clauses to generate the mission impact traces. Three different sets of interaction rules are created for mission impact analysis, including mission-task-service-host impact propagation rules, service impact propagation rules, and attack rules. The rules model the causality relations among facts. Finally, the graph generator can generate the mission impact graphs by parsing the mission impact traces.

Impact Current. To make effective cyber resilience recommendations, it is important to quantitatively measure the impact towards missions. Traditional security metrics mainly evaluate security risks at the asset level and lack the mission level assessment. Cheng et al. [5] explored a number of existing security tools and metrics, and presented new evaluation methods for cyber impact and mission relevance analysis. However, the approach presented lack a practical multi-layer graphical model to model the dependencies among entities at different abstract layers and support cross-layer mission impact assessment.

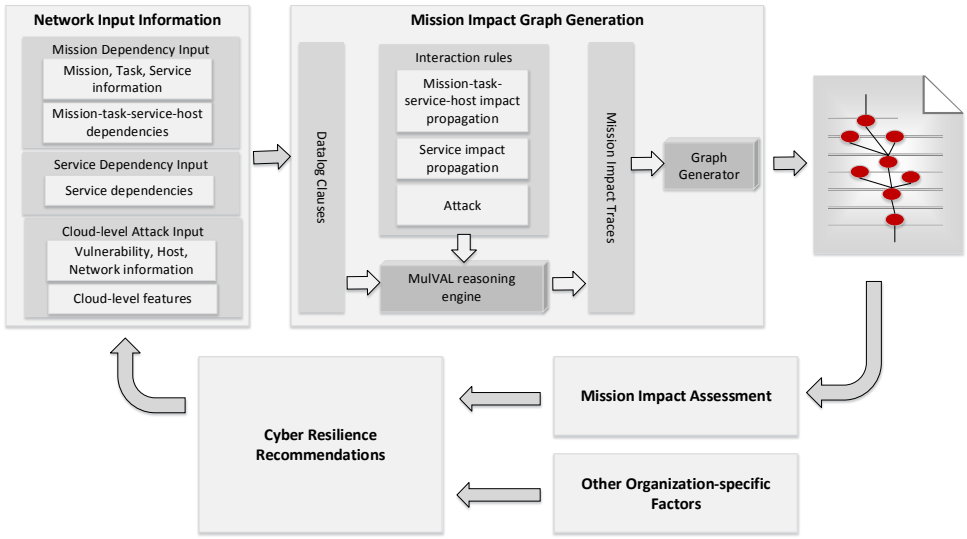

Figure 2. Mission Impact Assessment Framework.

We propose a mission impact metric named *impact current* on top of the mission impact graph model to assess the security risks of missions under certain network configurations. The impact current metric considers two factors: how difficult it is to impact a mission and how important the mission is. Therefore, the impact current metric is analogous to a circuit in which the mission importance has the effect of voltage and the impact resistance has the effect of electric resistance. The stronger the current is, the more impact risk the network has in terms of mission commitment.

While the mission importance is usually a predetermined value, the difficulty of impacting a mission depends on a number of aspects, such as network configurations, host information, and dependencies among entities, including missions, tasks, services and hosts. To measure the difficulty of impacting a mission, we use a metric called *impact resistance*. This metric is inspired by the attack resistance metric presented by Wang et al [6] to assess and compare the security of different network configurations. The computation of attack resistance is similar to computing the electric resistance of a series-parallel circuit. A larger resistance value implies increased security level of the network. However, the attack resistance metric only considers the exploit difficulty and is not mission-aware. We take the philosophy of attack resistance but extend it to mission impact graph model, which is quite different from the originally used exploit dependency graph in [6].

When applying the impact resistance metrics to mission impact graphs, the impact resistance value for root fact nodes are pre-assigned: determined by security experts or according to public vulnerability database *CVSS*. If the condition in a root fact node is easy to be leveraged by an attacker, the resistance value is low. For example, Node *netwrokServiceInfo()* has a low resistance because servers are meant to provide service and the likelihood for them to shut down or crash is very low. The resistance of nodes *vulExists()* can be assigned based on *the exploit difficulty* score associated with each vulnerability in CVSS database.

With the resistance value of root fact nodes, the resistance value of all remaining nodes in the mission impact graph can be computed. For each node, its precondition nodes can have two type of relationships, AND or OR. AND means that a node requires all precondition nodes being satisfied. OR represents that a node may have various ways of becoming true. Assuming a node m has a number of precondition nodes *1, 2, 3, ... n*, and we use IR_i to denote the impact resistance of node i, then the impact resistance of a derived node is the sum of all its preconditions' impact resistance if it's AND relation among the precondition nodes. That is,

$$IR_m = IR_1 + IR_2 + \cdots + IR_n$$

The impact resistance of a derived fact node is computed as follows if it's OR relation among precondition nodes:

$$\frac{1}{IR_m} = \frac{1}{IR_1} + \frac{1}{IR_2} + \cdots + \frac{1}{IR_n}$$

Based on the above models, every node in the mission impact graph is assigned an impact resistance value.

The impact current metric considers both the impact difficulty and the mission importance. Hence, the impact current value of a mission is determined by the mission importance and the impact resistance of the mission node. Assuming the mission importance for a mission m is IMP_m, and the impact resistance is IR_m, then the impact current IC_m is computed as:

$$IC_m = \frac{IMP_m}{IR_m}$$

If a mission is more important or it's easier to impact the mission, then the impact current value is bigger, which indicates higher mission impact risk.

Case Study

We've conducted a case study by applying the mission impact assessment framework towards a scenario with two enterprise networks in the same public cloud: A is a small online retail store, and B is a chemical research organization. A and B have their own missions, and they do not have any business relations with each other. In the attack scenario, the attacker can launch a multi-step attack to steal the project information on one of B's servers. We are able to generate the mission impact graph for this scenario and conduct mission impact analysis on top of the graph. By applying the impact current metrics, the impact current and impact resistance values of each node in the mission impact graph are calculated. The impact current value of a mission node is able to reflect both how difficult it is to affect a mission and how important the mission is. The impact current metrics are also used to assess the impact of condition changes and evaluate the resilience level of the networks. After changing some conditions (such as patching a vulnerability), we've recalculated the impact resistance and impact current values of each node in the mission impact graph. The analysis results show how a condition change may affect the impact towards missions. Although the impact of a condition change can be much more complicated in real enterprise networks, such analysis results can be used as one of the important references for security analysts to assess the impact of a condition change and make appropriate cyber resilience recommendations before the condition is actually applied.

Making cyber resilience recommendations is a delicate balancing act and a very complicated decision making process in real world. It is determined by many factors such as technical aspects, policies, resource limitations, and so on. The proposed mission impact framework, including the mission impact graph model and the impact current metrics, does not provide a complete solution to all the mission impact measurement and cyber resilience problems. It provides information for the security analysts to refer and help them better understand the status of the network and missions, but does not provide decisive suggestions. More resilience relevant factors can be incorporated into this framework to develop sophisticated cyber resilience solutions, which can even be customized for specific organizations or particular mission goals.

Disclaimer

This paper is not subject to copyright in the United States. Commercial products are identified in order to adequately specify certain procedures. In no case does such identification imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the identified products are necessarily the best available for the purpose.

References

[1] R. ROSS, R. GRAUBART, D. BODEAU, and R. MCQUAID, "Systems Security Engineering: Cyber Resiliency Considerations for the Engineering of Trustworthy Secure Systems," *Spec. Publ. NIST SP No 800-160*, vol. 2, 2018.

[2] X. Sun, A. Singhal, and P. Liu, "Towards Actionable Mission Impact Assessment in the Context of Cloud Computing," in *IFIP 31st Annual Conference on Data and Applications Security and Privacy (DBSec'17)*, 2017, pp. 259–274.

[3] "Proceedings of DARPA Information Survivability Conference and Exposition," Anaheim, California, 2001, vol. I & II.

[4] X. Ou, S. Govindavajhala, and A. W. Appel, "MulVAL: A Logic-based Network Security Analyzer.," in *USENIX Security Symposium*, 2005, vol. 8.

[5] Y. Cheng, J. Deng, J. Li, S. A. DeLoach, A. Singhal, and X. Ou, "Metrics of security," in *Cyber Defense and Situational Awareness*, Springer, 2014, pp. 263–295.

[6] L. Wang, A. Singhal, and S. Jajodia, "Measuring the Overall Security of Network Configurations Using Attack Graphs," in *Data and Applications Security XXI*, vol. 4602, S. Barker and G.-J. Ahn, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 98–112.