A Novel Centrality Measure for Network-wide Cyber Vulnerability Assessment

Arun V. Sathanur
Pacific Northwest National Labs
902, Batelle Blvd. Richland, WA 99352
Email: arun.sathanur@pnnl.gov
Phone: (509) 372-6578, FAX: (509) 375-6899

David J. Haglin
Pacific Northwest National Labs
1100 Dexter Ave N, Ste 400, Seattle WA 98109
Email: david.haglin@pnnl.gov
Phone: (206) 528-3263

***Abstract*—In this work we propose a novel formulation that models the attack and compromise on a cyber network as a combination of two parts - direct compromise of a host and the compromise occurring through the spread of the attack on the network from a compromised host. The model parameters for the nodes are a concise representation of the host profiles that can include the risky behaviors of the associated human users while the model parameters for the edges are based on the existence of vulnerabilities between each pair of connected hosts. The edge models relate to the summary representations of the corresponding attack-graphs. This results in a formulation based on Random Walk with Restart (RWR) and the resulting centrality metric can be solved for in an efficient manner through the use of sparse linear solvers. Thus the formulation goes beyond mere topological considerations in centrality computations by summarizing the host profiles and the attack graphs into the model parameters. The computational efficiency of the method also allows us to also quantify the uncertainty in the centrality measure through Monte Carlo analysis.**

***Index Terms*—Cyber Security, Risk Assessment, Vulnerability, Centrality, Random Walk Restart, Linear Solver, Graph Analytics**

I. Introduction

Cyber-security is a complex socio-technical problem requiring comprehension of multiple facets that include network protocol engineering, data mining and big graph-analytics, software engineering, network resilience, game theory and intent modeling. The increasing frequency, size and sophistication of cyber attacks have spurred a lot of research along the various constituents outlined above. Specifically, models and metrics that are designed to obtain an understanding of the vulnerability of enterprise networks have recently received a lot of attention.

In this work, we develop and implement novel metrics based on a probabilistic formulation to assess the host-to-host and network-wide risk proliferation emerging from the attack and spread of compromise on large cyber graphs by through multiple mechanisms. The formulation captures the mechanics of an attack/compromise scenario and its spread on the network by a probabilistic model and is versatile to allow for the incorporation of both the machine features and the behavior of the associated human users.

Our approach is distinct from existing vulnerability modeling approaches and can address the following:

Analyst Scenario

What is the risk potential of an enterprise laptop used by an employee who regularly clicks web-links in personal emails and downloads trial software from the Internet, on three specific high-asset enterprise workstations in particular and the enterprise as a whole?

Fig. 1. A potential analyst scenario involving risk assessment of a cyber network

- It captures in a very clear manner, the specific mechanisms that lead to cyber attack and compromise and the resulting risk propagation. Specifically, it allows for the incorporation of the roles of human behavior as well as the network structure in modeling the risk to the networks of interest as a whole. Therefore it can answer questions like the one presented in figure I in terms of probabilities.
- Allow for the incorporation of multiple factors, such as the host configuration profile, machine vulnerabilities, host network position and features involving the risky behaviors of the human users associated with the hosts
- Provide an analyst with a ranked list of hosts that have the highest potential for spreading the risk that can then inform the remedial strategies.
- The scalable nature of the formulation allows processing of massive graphs using the modern HPC resources
- The efficiency of the approach allows for the application of uncertainty quantification (UQ) techniques such as the Monte Carlo analysis in quantifying the spread in the vulnerability metrics resulting from the model parameter uncertainties.

We proceed to give a brief overview of the existing vulnerability / risk modeling approaches and describe our novel formulation based on random walk with restart for characterizing vulnerability and overall risk in an enterprise network.

II. Related work

Previous works on vulnerability modeling of enterprise networks have focused their attention along three main themes namely direct modeling of cyber graphs, modeling with attack graphs and finally Bayesian methods.

Pure graph-based approaches are normally grounded in applying concepts and metrics such as reachability, shortest paths and numerous modes of centrality computations to analyze vulnerabilities in a given network. Specifically the authors in Ref. [1] consider reachability concepts and propose a metric based on those to quantify exposure to vulnerability for attacks such as Pass-The-Hash. As a follow-up, the same authors examine a graph-coarsening approach in order to reduce the computational complexity of path-counting. In another recent work [2] , the authors consider metrics such as PageRank [3] and other centrality concepts to measure the badness of infrastructure elements in a network. The work also leverages graph pattern-mining concepts to study the evolution of cyber-threats. A number of pure graph-based approaches suffer from high computational complexity issues because of the presence of a large perimeter in the form of the Internet connected hosts and servers. Furthermore, pure graph-based approaches work based on topology without considering the underlying mechanics and the associated probabilities, that when included presents a more realistic picture.

Graph-theoretic concepts have been extended and extensively exploited and many metrics proposed in a slightly different context from those of the actual network graphs. Such approaches are based on the concept of attack graphs. Attack graphs not only consider the connectivity between the nodes but also model the states along which an attacker can move through by exploiting the local vulnerability between the machines. References [4]–[7] address a number of issues involving construction and traversal of attack graphs while also proposing various metrics in the process to quantify the vulnerability assessment. While attack graphs are exhaustive in their interpretation, they do not consider the probabilities of specific attacks i.e a relatively simple attack might be preferred over a complex attack. To address these shortcomings and to move beyond mere counting of the vulnerabilities, probabilistic and Bayesian versions of attack graphs have been proposed and analyzed [8]. While the work presented in Ref. [9] was one of the first to bring Bayesian Network concepts to attack graphs, the authors in ref. [10] build on those concepts and relax the independence assumption between various attack paths in Bayesian attack graphs.

Attack graphs, even though they are exhaustive in their enumeration of vulnerability paths suffer from high computational complexity because of explosion in the state space. As a result many approaches have been proposed to reduce the complexity of the method. While reference [11] explores using influence metrics such as the PageRank for pruning the attack graph size, the works in [12] explore improving attack graph visualization by pruning out portions that are not relevant and by aggregating steps that are related.

Beyond attack graphs, other probabilistic methods that have received attention to quantify vulnerability in a cyber-attack scenario include the usage of Hidden Markov Models (HMM) [13] and derivation of metrics similar to Mean Time To Failure by using Markovian models [14]. Additionally game-theoretic methods [15], [16] have been recently investigated to understand the payoffs involved in the case of a cyber-attack scenario. With this background, we now present our probabilistic formulation for quantifying network-wide cyber-vulnerability and risk.

III. Formulation

We first develop a probabilistic local model of vulnerability around a specific node in the cyber-graph and then scale the same to a large network [17]. For that we specifically identify a set of two main mechanisms that can lead to a host being compromised, as illustrated in figure 2.

Direct : The host under consideration (large orange node) can be infected with malware via an email attachment or by visiting a malicious website or via an infected USB stick or through stolen credentials.

Network : A host that is in the neighborhood of the host under scrutiny is first compromised (one of the small blue nodes) through any of the two mechanisms and then the host under consideration is compromised through a network service.

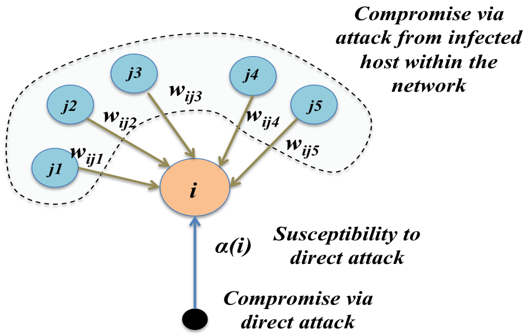
Compromise via attack from infected host within the network
j3
j4
j2
j5
j1
w_{ij3}
w_{ij2}
w_{ij4}
w_{ij5}
w_{ij1}
i
α(i)
Susceptibility to direct attack
Compromise via direct attack

Fig. 2. A concise representation of direct and indirect modes of vulnerability along with the local model parameters.

Within the network of interest, consider an arbitrary host abstracted by node i along with the set of connected nodes from which the node i is reachable in one hop. Let $p^T_{cmp}(i)$ denote the total probability of the host i being compromised. Now the total probability of compromise $p^T_{cmp}(i)$ can be broken into two parts namely the probabilities of compromise via the direct and the network modes as discussed above. Thus we can write:

$$p^T_{cmp}(i) = \alpha(i)p^D_{cmp}(i) + \beta(i)p^N_{cmp}(i) \tag{1}$$

$p^D_{cmp}(i)$ denotes the probability of compromise for node i through a direct attack and $p^N_{cmp}(i)$ denotes the probability of compromise for node i via attack from within the network. The parameters $\alpha(i)$ and $\beta(i)$ denote the susceptibilities to direct attacks and to attacks from within the network respectively. Therefore we have $\alpha(i) \geq 0$, $\beta(i) \geq 0$ and $\alpha(i) + \beta(i) \leq 1$. By assuming that the hosts are always susceptible to either of the two mechanisms of compromise, we set $\beta(i) = (1 - \alpha(i))$.

This assumption is being relaxed in the ongoing work. Now for the network contribution itself we can write:

$$p^N_{cmp}(i) = \sum_{j,(j,i)\in\mathcal{E}} w_{ij} p^T_{cmp}(j) \tag{2}$$

Therefore we have:

$$p^T_{cmp}(i) = \alpha(i) p^D_{cmp}(i) + (1-\alpha(i)) \sum_{j,(j,i)\in\mathcal{E}} w_{ij} p^T_{cmp}(j) \tag{3}$$

Here $\mathcal{E}$ denotes the set of directed edges on the cyber-graph under consideration. The coefficient w_{ij} denotes the probability that the attack on node i comes from its immediate neighbor node j once j has been compromised. If we look back from the point of view of node i, we can see that $\sum_{j,(j,i)\in\mathcal{E}} w_{ij} = 1$. Further details about the model parameter estimation are outlined in section IV.

Finally scaling the formulation given by equation 3 to the entire network with N nodes, we obtain a matrix-vector equation as follows.

$$\boldsymbol{p^T_{cmp}} = \boldsymbol{\alpha p^D_{cmp}} + ((\boldsymbol{I} - \boldsymbol{\alpha W}))\, \boldsymbol{p^T_{cmp}} \tag{4}$$

Here $\boldsymbol{p^T_{cmp}}$ and $\boldsymbol{p^D_{cmp}}$ are vectors of size $N \times 1$ denoting respectively the total probability of compromise and probability of compromise through direct attack for all the nodes on the network. $\boldsymbol{I}$ denotes the identity matrix of size $N \times N$. $\boldsymbol{\alpha}$ denotes the diagonal matrix with entries corresponding to the susceptibility to direct attack for each of the hosts on the network. Note that the modeling of direct compromise on each of the nodes through the quantity $\boldsymbol{p^D_{cmp}}$ allows us to model only the enterprise network and not consider the large perimeter (in the form of internet nodes) outside of the network. Finally $\boldsymbol{W}$ denotes the sparse, stochastic weight matrix with entires given by the w_{ij}s discussed earlier.

This allows us to express the total probabilities of compromise as a transfer matrix times the vector of probability of compromise through a direct attack.

$$\boldsymbol{p^T_{cmp}} = \boldsymbol{V p^D_{cmp}}; \boldsymbol{V} = (\boldsymbol{I} - (\boldsymbol{I} - \boldsymbol{\alpha})\boldsymbol{W}))^{-1} \boldsymbol{\alpha} \tag{5}$$

If we think of the vector of probabilities denoting compromise via direct attack, $\boldsymbol{p^D_{cmp}}$ as some sort of a source signal eq. 5 lets us predict the impact of such a source signal on the entire network in terms of compromisability [17]. Note that all possible paths between *source* and *destination* nodes are automatically taken care of in the formulation. Following the work of Chung and Yao [18], matrix $\boldsymbol{V}$ is identified as a form of discrete Green's function which is grounded in the RWR paradigm as opposed to the usage of the random-walk Laplacian in the original work [17]. Thus V_{ij} is a measure of directed proximity or relevance between the nodes j and i along all possible paths and not just the shortest path. Such a random walk based approach is more suited in this scenario where the malicious actor doesn't have the knowledge of the complete network and proceeds from node to node via exploration of vulnerability. The same is reflected in the assignment of the weights which is grounded in the vulnerability attributes of the connected hosts that define the edges. We further note that because the matrix $\boldsymbol{W}$ is a stochastic matrix, the matrix $(I - (\boldsymbol{I} - \boldsymbol{\alpha})\boldsymbol{W}))$ is diagonally dominant and has all the non-diagonal entries negative. Thus it is an *M-Matrix* guaranteed to have an *all-positive* inverse [19]. To quantify the impact of the compromise through a direct attack on a given node say node i on the network we set $p^D_{cmp}(j) = 0; j \neq i$. Thus the quantity $\left(\sum_{j=1}^N V_{ji}\right) p^D_{cmp}(i)$ represents the expected number of hosts compromised by the attack on node i. We term this *amplification factor* as *vulnerability centrality*.

$$C_V(i) = \left(\sum_{j=1}^{N} V_{ji}\right) \tag{6}$$

It can be easily seen that the vulnerability centrality metric $\boldsymbol{C_V}$ can be directly computed as a linear solve without the need for the matrix inversion.

IV. Model Parameters

The coefficient $\alpha(i)$ depends primarily on attributes such as the position of the node on the network, the extent of exposure to the internet, the kind of programs and services that run on the host that the node abstracts and even the behavior of the associated human users. Thus if we have two scores $s_d(i)$ and $s_n(i)$ that represent the extent to which a host associated with node i is vulnerable to compromise via direct attack versus that of an attack from a compromised host within the network, we can write $\alpha(i) = \left({}^{s_d(i)}/_{(s_d(i)\,+\,s_n(i))}\right)$. Qualitatively it follows that a laptop host computer used by a human user within and outside an enterprise has a greater α than a desktop within an enterprise. Similarly a router within an enterprise network can potentially have $\alpha(i) \rightarrow 0$.

The w_{ij}s on the other hand denote in some sense how the services and ports line up in a way so as to be able to compromise i while attacking from j. Illustrations of these mechanisms in the context of attack graphs is provided in an earlier work by Jajodia and others [6]. We could formalize the computation of the $w^{'}_{ij}$ (un-normalized version of w_{ij}) as a cosine similarity between a "*service vector*" $\boldsymbol{s^j_w}$ on the host associated with node j and a "*vulnerability vector*" $\boldsymbol{v^i_w}$ for the target host associated with node i. For example if the malicious actor can potentially use an *ftp* program to connect from j to i which has an open *ftp* port we get a match thereby contributing to the $w^{'}_{ij}$. Multiple attributes can then be included as part of the computation by defining $w^{'}_{ij} = \boldsymbol{s^j_w} \cdot \boldsymbol{v^i_w}$. The final w_{ij}s are then computed by normalizing the sum of the $w^{'}_{ij}$s to unity.

To summarize, the model parameters corresponding to the $\boldsymbol{\alpha}$ vector are dependent on the host profiles, the host position on the cyber-graph and the risky behavior associated with the human users. The model parameters corresponding to the matrix $\mathbf{W}$ depends on the existence of local vulnerabilities between a pair of connected hosts either at the application-level or through the network connectivity.

The Common Vulnerability Scoring System (CVSS) [20], [21], now in its third version, is an open industry standard that assigns vulnerability scores to systems based on the aggregation of several metrics that quantify the construction and impact of exploits. CVSS scores can adequately characterize the vulnerabilities and tools such as k-core decomposition can characterize the position on the graph itself. However modeling the risky nature of the human behavior is not that straightforward. Some aspects of the user-behavior data might be inferred from the NetFlow data. Alternatively, in the absence of concrete information, the contribution of the risky human behavior to the coefficient α can be modeled in a *fuzzy* manner through the use of distributions over an interval. This provides the motivation for the computation of the spreads in the centrality metrics through UQ techniques such as Monte Carlo analysis. These discussions allow us to propose a workflow for the risk assessment and mitigation in an enterprise scenario as depicted by figure 3.

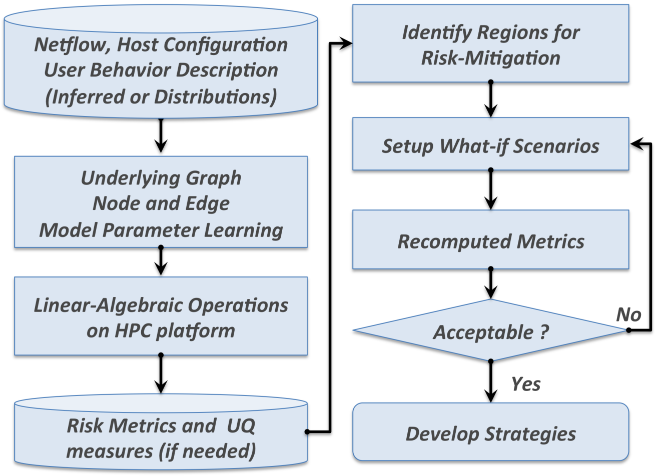

Fig. 3. Proposed flow depicting the how the vulnerability assessment can be employed in an enterprise scenario

V. Experiments

In this section we consider two examples. The first is a toy example with seven nodes as shown in fig. 4 while the second is a random graph with 1000 nodes. With respect to the toy network in fig. 4, based on their roles in the network and the potential similarities between the machines, values are assigned to the $\boldsymbol{\alpha}$ vector and $\boldsymbol{W}$ matrix entries as depicted below.

$$\boldsymbol{\alpha} = \begin{bmatrix} 0.05 & 0.08 & 0.12 & 0.18 & 0.55 & 0.8 & 0.2 \end{bmatrix}$$

$$\boldsymbol{W} = \begin{bmatrix} 0 & 0.6 & 0.1 & 0.2 & 0.1 & 0 & 0 \\ 0.8 & 0 & 0 & 0 & 0 & 0.1 & 0.1 \\ 0.3 & 0 & 0 & 0.7 & 0 & 0 & 0 \\ 0.6 & 0 & 0.2 & 0 & 0.2 & 0 & 0 \\ 0.2 & 0 & 0 & 0.8 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

The bar chart in Fig. 5 shows the various vulnerability centralities. We found that nodes that have a high probability of direct attack namely H_2 and H_3 have the highest vulnerability centrality. Node S_1 also figures high on the list despite having a low α because of the higher weighted out-degree. These observations are compatible with the first order approximation for the vulnerability centrality from eqs. 5 and 6. Note that the topologically central nodes R_1 and R_2 figure low on the list because of low probabilities of direct compromise (very low α values).

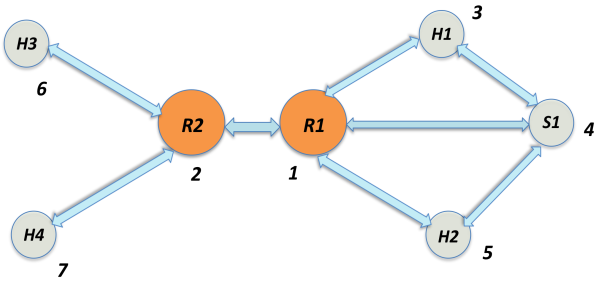

Fig. 4. A toy cyber network with two routers (R_1 and R_2), a server (S_1) and four hosts ($H_1 \ldots H_4$). The communication paths between them are shown as well. Hosts H_2 and H_3 are laptops routinely used outside the enterprise while H_1 and H_4 are desktops with in the enterprise.

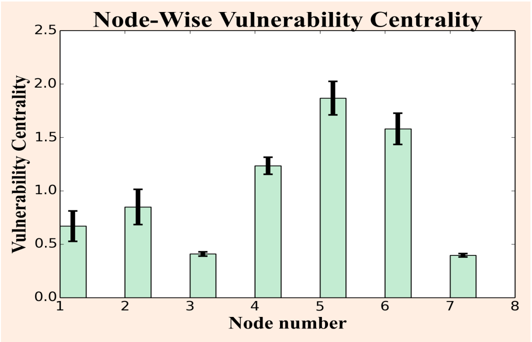

Fig. 5. A bar chart showing the influence of each of the nodes on the network in terms of vulnerability centrality. The error bars represent the effect of uncertainty as estimated from a 1000 run Monte Carlo analysis. Some of the $\boldsymbol{\alpha}$ values are made uncertain and they are modeled as following an uniform distribution within a given interval.

Figure 6 shows the effect of model parameter uncertainty on the $top-k$ rankings of nodes on a cyber-graph in terms of the vulnerability centrality metric. We applied variations to the α values of each of the nodes in an Erdős Rényi random graph of 1000 nodes and edge probability of 0.03. Each α_i is drawn from its own independent uniform distribution with 10% and 20% perturbations from the baseline case. We compute and plot the histogram of the Spearman Rank Correlation Coefficient (ρ) between the $top-100$ nodes in the unperturbed case with the same nodes in each of the 1000 Monte Carlo runs. Clearly the fact that correlations are not all close to 1, implies that even the rankings are significantly affected by the model parameter variations. Further, as expected, for larger variations, the deviation of ρ from 1 is larger and the spread itself becomes larger. Intuitively, this means that even though a host is reasonably secured in terms of its position on the cyber graph and possessing a low vulnerability profile, a larger than expected α value by virtue of the risky behavior on the part of the associated human user (such as access through mobile devices via non-secure networks or visiting malicious websites) could potentially increase the vulnerability centrality of the associated machine. Thus the characterization of the spread in the centrality measures is a very important step in the flow depicted in fig. 3.

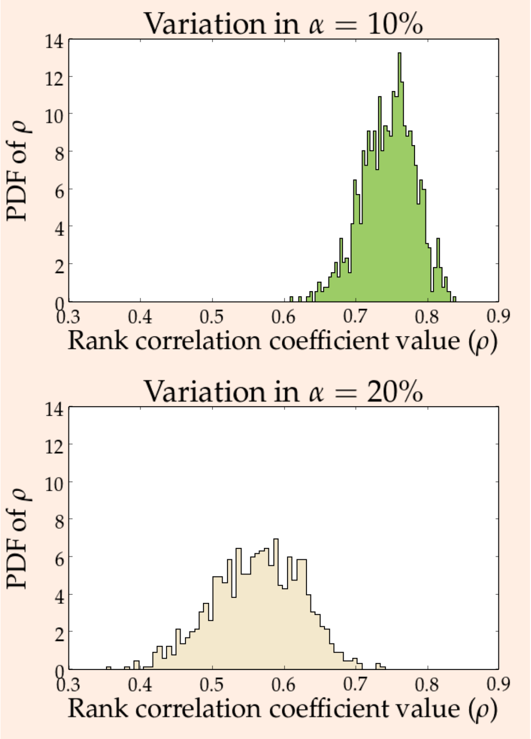

Fig. 6. PDF of ρ computed with respect to the rankings of the top-100 nodes of the unperturbed case and the same nodes for each of the MC runs. Top: A 10% variation in the model parameter α and Bot: A 20% variation in the model parameter α

VI. Concluding remarks and Future work

IT analysts are often overwhelmed with large amounts of information related to threats and vulnerabilities spread across many different metrics and tools. This task is a step towards integrating the various mechanisms of vulnerability including the crucial human-user aspect. The result is a set of easily understandable metrics that probabilistically quantify the risk either to a specific set of machines or to the network as a whole that can be utilized by analysts and the management in decision-making. By capturing the mechanisms probabilistically, stronger paths of compromise spread and more risky use-behaviors are automatically highlighted thereby reducing the information overload to the analysts. The presented metrics, together with the associated asset information, allow for the identification of the regions of the network that require greatest effort in terms of risk mitigation. This includes targeted education of human users on enforcing the best practices of cyber-defense.

Ongoing work involves completing the model parameter estimation, applying the concepts to real-world cyber-graphs and development of computationally efficient methods for uncertainty quantification.

Acknowledgement

This research was supported by the High Performance Data Analytics program at the Pacific Northwest National Laboratory (PNNL). PNNL is a multi- program national laboratory operated by Battelle Memorial Institute for the US Department of Energy under DE-AC06- 76RLO 1830.

References

[1] J. R. Johnson, E. Hogan *et al.*, "A graph analytic metric for mitigating advanced persistent threat," in *Intelligence and Security Informatics (ISI), 2013 IEEE International Conference on*. IEEE, 2013, pp. 129–133.

[2] A. Boukhtouta, D. Mouheb, M. Debbabi, O. Alfandi, F. Iqbal, and M. El Barachi, "Graph-theoretic characterization of cyber-threat infrastructures," *Digital Investigation*, vol. 14, pp. S3–S15, 2015.

[3] A. N. Langville and C. D. Meyer, *Google's PageRank and beyond: The science of search engine rankings*. Princeton University Press, 2011.

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

[5] S. Noel, S. Jajodia, L. Wang, and A. Singhal, "Measuring security risk of networks using attack graphs," *International Journal of Next-Generation Computing*, vol. 1, no. 1, pp. 135–147, 2010.

[6] S. Jajodia and S. Noel, "Topological vulnerability analysis," in *Cyber Situational Awareness*. Springer, 2010, pp. 139–154.

[7] J. Homer, S. Zhang, X. Ou, D. Schmidt, Y. Du, S. R. Rajagopalan, and A. Singhal, "Aggregating vulnerability metrics in enterprise networks using attack graphs." *Journal of Computer Security*, vol. 21, no. 4, pp. 561–597, 2013.

[8] A. Singhal and X. Ou, *Security risk analysis of enterprise networks using probabilistic attack graphs*. Citeseer, 2011.

[9] Y. Liu and H. Man, "Network vulnerability assessment using bayesian networks," in *Defense and Security*. International Society for Optics and Photonics, 2005, pp. 61–71.

[10] M. Frigault and L. Wang, *Measuring network security using bayesian network-based attack graphs*. IEEE, 2008.

[11] V. Mehta, C. Bartzis, H. Zhu, E. Clarke, and J. Wing, "Ranking attack graphs," in *Recent advances in intrusion detection*. Springer, 2006, pp. 127–144.

[12] J. Homer, A. Varikuti, X. Ou, and M. A. McQueen, "Improving attack graph visualization through data reduction and attack grouping," in *Visualization for computer security*. Springer, 2008, pp. 68–79.

[13] A. Årnes, F. Valeur, G. Vigna, and R. A. Kemmerer, "Using hidden markov models to evaluate the risks of intrusions," in *Recent Advances in Intrusion Detection*. Springer, 2006, pp. 145–164.

[14] M. Dacier, Y. Deswarte, and M. Kaâniche, "Models and tools for quantitative assessment of operational security," in *Information systems security*, 1996.

[15] S. Roy, C. Ellis, S. Shiva, D. Dasgupta, V. Shandilya, and Q. Wu, "A survey of game theory as applied to network security," in *System Sciences (HICSS), 2010 43rd Hawaii International Conference on*. IEEE, 2010, pp. 1–10.

[16] S. Chatterjee, M. Halappanavar, R. Tipireddy, M. Oster, and S. Saha, "Quantifying mixed uncertainties in cyber attacker payoffs," in *Technologies for Homeland Security (HST), 2015 IEEE International Symposium on*. IEEE, 2015, pp. 1–6.

[17] A. V. Sathanur, V. Jandhyala, and C. Xing, "Physense: Scalable sociological interaction models for influence estimation on online social networks," in *IEEE International Conference on Intelligence and Security Informatics*. IEEE, 2013, pp. 358–363.

[18] F. Chung and S.-T. Yau, "Discrete green's functions," *Journal of Combinatorial Theory, Series A*, vol. 91, no. 1, pp. 191–214, 2000.

[19] A. Greenbaum, *Iterative methods for solving linear systems*. Siam, 1997, vol. 17.

[20] "The common vulnerability scoring system," https://www.first.org/cvss, [Online].

[21] K. Scarfone and P. Mell, "An analysis of cvss version 2 vulnerability scoring," in *Proceedings of the 2009 3rd International Symposium on Empirical Software Engineering and Measurement*. IEEE Computer Society, 2009, pp. 516–525.