Performance Analysis of Scalable Attack Representation Models

Jin Hong, Dong Kim

Performance Analysis of Scalable Attack Representation Models

Jin B. Hong and Dong Seong Kim

Computer Science and Software Engineering,
University of Canterbury,
Christchurch,
New Zealand
jho102@uclive.ac.nz,dongseong.kim@canterbury.ac.nz

Abstract. Attack graphs (AGs) have been widely used for security analysis. The construction of the graph-based attack models including the AG have been studied, but the security evaluation considering the full attack paths cannot be computed using existing attack models due to the scalability problem. To solve this, we propose to use hierarchical attack representation models (HARMs). First, we formulate key questions that need to be answered to compare the scalability of existing attack models. We show the scalability of the HARMs via simulations, by taking into account practical attack scenario based on various network topologies.

Keywords: Attack Graph, Attack Tree, Complexity Analysis, Security Model, Scalability

1 Introduction

Attack models are used to evaluate the security of networked systems and to provide countermeasures to enhance the security [1–9]. Previous studies showed that the graph-based attack models (e.g., attack graph (AG) [10], multiple prerequisite graph (MPG) [11], two-layered attack graph (TLAG) [4]) have a scalability problem if full attack paths are considered [10–14]. The tree-based attack models (e.g., attack tree (AT) [15], attack countermeasure tree (ACT) [16]) can be constructed and evaluated in a scalable manner depending on their structures. Methods to construct tree-based attack models are described as either decomposition of the attack goal [7], computing min-cuts from the networked system with an assumed attacker and the target [8], or drawn by security experts manually. As far as we know, there is no automated generation method that captures all possible attack paths in tree-based attack models.

There are phases in the lifecycle of the attack models [17]. The pre-processing phase gathers the network and security information, the construction phase generates the attack model, the evaluation phase processes the security analysis using security metrics, and the modification phase captures any updated events in the networked system and modifies the attack model accordingly.

Previous researchers proposed various attack model structures that improved the scalability in the construction phase, and heuristic methods, such as graph simplifications, are used to avoid the scalability problem in the evaluation phase [4, 5, 11]. Full

attack paths contain all possible attack scenarios, and analysing full attack paths can drive one to find out the optimal security solution. But as far as we know, computing full attack paths to evaluate the security of networked systems is scoped to small networks only, because of the scalability problem. Existing attack models are not scalable to compute the full attack paths in the networked system, because there are many attack paths in a general networked system (e.g., a networked system with fully connected components). Therefore, we require a general solution to the scalability problem for the attack models.

We proposed hierarchical attack representation models (HARMs) to improve the scalability problem [17]. Our previous study shows that the HARMs have better or equal complexities in three phases of the attack models, such as construction, evaluation and modification, compared with an AG and an AT. It is important that the attack models are scalable for all network topologies (e.g., mesh, star, complete), so that all types of the networked systems can be modelled, analysed and secured (e.g., smart grids, sensor networks, ad hoc networks). However, our previous study only considered the worst case analysis using the system where nodes are fully connected.

We denote an AG that only represents the network structure (i.e., the full attack paths information is not expressed) as a simplified AG. The simplified AG representation is used in the layers of the HARMs, compared with the simplified AG in the phases of construction and evaluation. The simulation considered different network topologies and variable number of vulnerabilities to improve the limitations of simulations observed in previous works. We consider the evaluation phase to compute the full attack paths. The contributions of this paper are:

- To list key questions to compute the scalability, and identify unanswered questions for existing attack models;
- To simulate the attack model's construction and evaluation phases, and compare the scalability considering multiple vulnerabilities and various network topologies using a practical network system

The rest of the paper is as organised as follows. In Section 2, related work is introduced. In Section 3, the HARMs and existing attack models are compared using five key questions. Section 4 represents the simulation result, and discussion is given in Section 5. Finally, section 6 concludes this paper.

2 Related Work

Over the last decade, many attack models have been proposed. There are no general tree-based attack model construction methods that avoid the scalability problem. Hence, we will consider graph-based attack models to compare the performance in the phases of attack models. Sheyner [10] used a full attack graph, but it had a scalability problem. Many researchers presented efficient methods of the AG construction and evaluation. To improve the efficiency of the attack models, researchers considered improvements on the full AG [18–20], or proposed new graph-based attack model structures [4, 5, 11].

Ou *et al.* [5] used a logical attack graph (LAG), and the construction of the LAG can be done in a time of polynomial complexity. However, evaluation method and its

complexity analysis are not mentioned. The simulation for the construction phase assumed that each host has same vulnerabilities, but in real systems we can expect various number of vulnerabilities for different hosts with different services and applications. Moreover, exploiting vulnerability does not necessarily give the root privilege, as assumed in their work. Ingols *et al.* [11] used a predictive graph to avoid the scalability problem of the AG. Later, they proposed a multiple prerequisite graph (MPG), more scalable than the predictive graph. They reported the scalability of the MPG has the size complexity of O($n$log$n$), where n is the number of hosts in the networked system. They used heuristic methods to simply the MPG to evaluate the network security. A direct attack was used for the attack scenario in the simulation. Although they used more than one type of vulnerabilities, the number of vulnerabilities was fixed. Xie *et al.* [4] used a two-layer attack graph (TLAG), where the upper layer captured the host reachability, and the lower layer captured the vulnerability information. There are similarities between the HARMs and the TLAG, but the TLAG stores the lower level information in each edge (i.e., construct the vulnerability attack graph between host pairs), but the HARMs store the lower level information in each host. As a result, less memory space is required for the HARMs than the TLAG, as well as construction and evaluation times in general. The network structure was described, but the simulation did not perform the scalability test, and the vulnerability information was not given.

We compare the structure of the HARMs, which are designed to use any attack models in their layers, with simplified AG, LAG, MPG, and TLAG in terms of scalability by taking into account the worst case performance.

3 Model Comparisons

Existing attack models and their studies lack in comparative studies to show how well their models scale in various environments and attack scenarios. We listed five key questions to compare the scalability of attack models:

Q1 Was the computational complexity analysis performed?
Q2 Compared against other attack models?
Q3 Different network topologies have been considered?
Q4 The Effect of variable number of vulnerabilities for hosts is considered?
Q5 Different types of vulnerabilities (user/root) are considered?

To compare the scalability of the HARMs, we will consider different graph-based attack models (simplified AG, LAG, MPG, and TLAG) and compare their scalability in the construction and the evaluation phase by inspecting their model structures and features. We assume the reachability information is given as in [11]. Also, we assume that other information (e.g., credentials, interfaces, ports) is abstracted in the attack model, as they are linearly proportional to the number of hosts and vulnerabilities. The HARMs will consider the simplified AG model in both the upper and the lower layer. In the evaluation phase, we will consider the calculation of the full attack paths. We will only consider the number of hosts and vulnerabilities of each attack model to compare the scalability, because they are the major variable factors for the scalability among many others.

3.1 The Construction Phase

Studies on existing attack models have failed to answer some of the key questions when analysing the scalability of attack models. The answers for the key questions considering the construction phase are given in Table 1. Moreover, the corresponding attack model is required to be modified if there is a change in a networked system. The existing attack models that are not in a hierarchical representation require inspecting all attack model components to make modifications accordingly. However, attack models using the hierarchical representation (e.g., the HARMs and the TLAG) may apply modifications in the required layer only.

Table 1. Studies covered for the construction phase

Attack models	TLAG [4]	LAG [5]	MPG [11]	HARMs [17]
Q1	Yes	Yes	Yes	Yes
Q2	No	Yes	Yes	Yes
Q3	No	Yes	No	Yes
Q4	No	Yes	No	Yes
Q5	No	No	Yes	Yes

The construction phase of the attack model is required to retrieve the network information and connecting the network components specific to the attack model requirements (e.g., connecting a vulnerability node to its subsequent vulnerabilities or hosts based on the reachability, application and port information). We assume that the vulnerabilities of each host can be exploited based on the reachability information only. The analysis is focused on answering the key questions, and identifying key features of each attack model.

The construction of the full AG requires the calculation of full attack paths in the construction phase, which has a scalability problem that is impractical for a large (sized) networked system [12]. Instead, a simplified AG can be constructed, which is a simplified version of the full AG that only captures the network properties. There are other simplified versions of the full AG, which are modified to fit their usage [21]. The connections between vulnerabilities and hosts are independent in the simplified AG, so there are more edges in the simplified AG than the HARMs. However, the computational complexity of the construction phase of simplified AG and the HARMs is equivalent [17].

The LAG has a construction complexity of O(δN), where N is the number of hosts in the networked system and δ is the time to find the host in the lookup table [5]. However, they assumed all vulnerabilities are the same (remote to exploits). Since each derivation node is an *AND* node, repeated nodes are required for each exploit if there are multiple sources it could be exploited from. If we allow the derivation nodes to be *OR* nodes, the number of repeated nodes will be reduced.

The MPG graph has the number of components linearly proportional to the number of hosts and vulnerabilities in the networked system [11]. Their performance in the simulation showed almost linear relationship between the computational time and

the number of hosts. Also, they use *prerequisite* nodes in the model. This reduces the number of independent connections between hosts and vulnerabilities. But in the worst case, there will only be a single reachability group without the number of edges reduced. Moreover, they reported that even after 99% reduction in the graph size, they still had a problem representing the MPG because of complex relationship between hosts and vulnerabilities. In work [6], they described the client-side attacks using the reverse reachability calculations as an additional function.

The TLAG divides the network into two layers, where the upper layer captured the host reachability and the lower layer captured the vulnerability information between each host pair [4]. The main representation is simplified AG for both the upper and the lower layer. The analysis on the computational cost was given as $O(n^2)$, where n is the number of hosts in the network. The number of vulnerabilities is not assessed in the complexity analysis. They assumed that only the user level access is enough to compromise the host, and they did not take into account super-user (i.e., the root privilege). The lower level construction is based on host pairs. If the network consists of many edges between hosts, then the number of lower layer models increases proportional to the number of edges, which can be up to $O(n^2)$ number of edges. However, if the lower layer models in the TLAG are identical between different host pairs, they can share the same lower layer model. In the optimal case, where a host will have the same exploit sequence from any source, the number of lower layer models is linearly proportional to the number of hosts in the TLAG. Only the optimal case will show the same number of lower layer models with the HARMs. However, the TLAG requires that every edge has a reference to its lower layer information, whereas the HARMs only require the reachability information.

3.2 The Evaluation Phase

The evaluation process is a critical part in the security analysis, but due to the structural design, some attack models lack in efficient security analysis (e.g., scalability problem). Studies on existing attack models only answered a few of key questions listed. This is shown in Table 2.

Table 2. Studies covered for the evaluation phase

Attack models	TLAG [4]	LAG [5]	MPG [11]	HARMs [17]
Q1	Yes	No	Estimated	Yes
Q2	No	No	Yes	Yes
Q3	No	No	No	Yes
Q4	No	No	No	Yes
Q5	No	No	Yes	Yes

We consider the evaluation phase to compute the full attack paths (i.e., all possible attack sequences). Existing attack models use simplifications and heuristic methods (e.g., graph simplification [11]) to evaluate the network security, but they only consider specific attack scenarios and subset of all possible attacks. Matrix evaluation can be

used to compute the overall security of the networked system, but it lacks in detailed analysis of the individual attack path. To improve these limitations, we compute the full attack paths.

Security analysis using the LAG is not available [5]. We estimate the evaluation complexity for the LAG is equivalent to the simplified AG, because each fact node (or also known as a host node with a given privilege) makes an independent connection to derivation nodes (or also known as vulnerability nodes). The number of paths from each fact node increases exponentially as the number of choices increases in the attack path, which are the same characteristics found in the simplified AG.

The evaluation of the MPG is to simplify the graph, then analyse the security. The evaluation complexity is estimated from the trend observed in their simulation. However, we will consider computing the full attack paths. The number of edges in the MPG depends on the number of reachability groups. If we consider the worst case (i.e., a complete graph), the performance of the MPG is equivalent to the simplified AG with a single prerequisite node (i.e., a single reachability group). If there are multiple prerequisite nodes, then connections between hosts and vulnerabilities are grouped by the prerequisite nodes, and it reduces the complexity in the evaluation. The optimal number of reachability groups is not analysed. Their analysis or simulation did not consider different network topologies or variable number of vulnerabilities.

The evaluation of the TLAG considered the overall security using the probability of an attack. But this evaluation method lacks in assessing different attack paths and their effects. The number of host-pair attack graphs (i.e., lower layer information) was not linearly proportional to the number of hosts. The analysis did not consider different network topologies, variable number of vulnerabilities in their analyses and simulations, and vulnerabilities giving different privileges when exploited.

The evaluation of the HARMs was obtained using the simplified AG in both the upper and the lower layer, but we observe that the computational complexity in both construction and evaluation phase have improved in comparison to the simplified AG. The improvements achieved using the HARMs compared with the simplified AG will be shown in the next section.

4 Simulation Result

The HARMs improve the efficiency of the attack model by reducing the number of independent connections between hosts and vulnerabilities. We investigate the improvements achieved using the HARMs through simulations. Our simulation setup used identical hosts, so that the scalability of the HARMs and the TLAG will be identical. We will consider a network with heterogeneous nodes in our future work to compare the scalability of the HARMs and TLAG.

The result of the simulation must be credible using appropriate quantification methods. For our simulations, we used an automated network simulation tool named *Akaroa2*, which produces credible stochastic simulation results with statistical analysis [22, 23]. All simulation results were obtained with the confidence level of 0.95, and the relative error of 0.05. The simulation program was coded using Python, and it was conducted in a Linux environment with Intel(R) Core2 Quad CPU 2.66GHz with 3.24GB of RAM.

4.1 A Practical Network Structure

In our previous study [17], we considered a fully connected system for the complexity analysis. The attack scenario used in the simulation is similar to that in the experiment conducted by Ingols *et al.* [11]. The networked system used in the simulation is shown in Figure 1. A network in our simulation setup used four sites, with each site consisting of five DMZ hosts, five administrative LAN hosts, and ten internal subnets. Each subnet has a bus topology to connect all hosts. The port information and the firewall rules are abstracted. We assigned ten remote-to-other vulnerabilities to half of hosts in each subnet, and the other half with one remote-to-root and nine remote-to-other vulnerabilities. The attack scenario was to compromise a host in the DMZ, an administrative LAN host, and all hosts in the network that has a remote-to-root vulnerability. Hosts that were not directly reachable from the attacker were compromised using other hosts as a stepping stones. The number of hosts in each subnet was increased to compare the scalability between the simplified AG and the HARMs. The scalability comparison is shown in Figure 2 for construction, and 3 for evaluation.

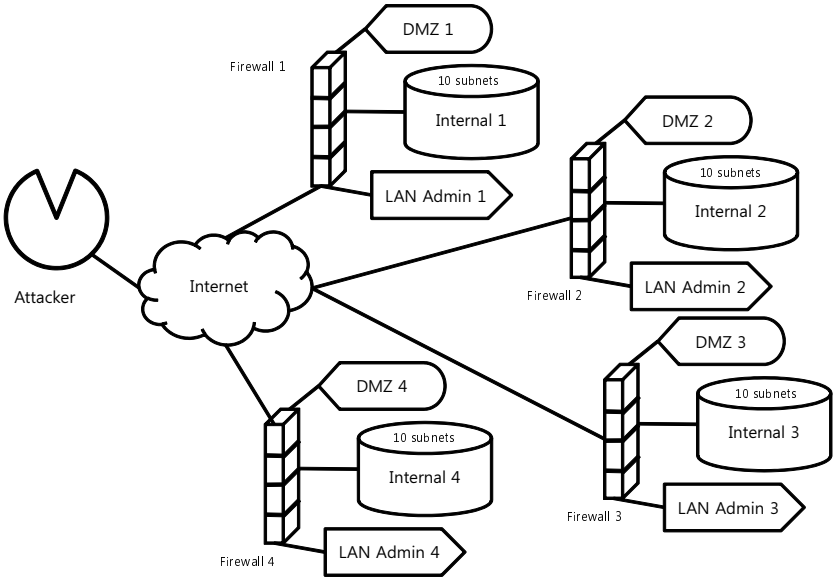
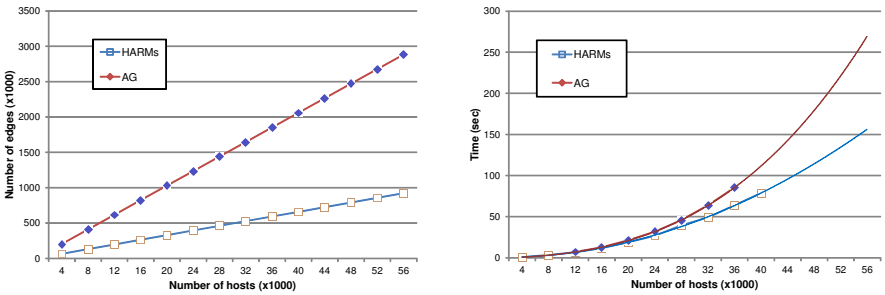


Fig. 1. A Networked System Configuration for Simulation

Figure 2 shows the performances of the simplified AG and the HARMs in the construction phase. Only Ingols *et al.* [11] and our work considered different types of vulnerabilities in the simulation. The simulation result shows that the number of edges in the simplified AG increases more rapidly than the HARMs. However, construction times for the simplified AG and the HARMs do not have a significant difference. This indicates that the number of edges has a little influence on the construction time. Both

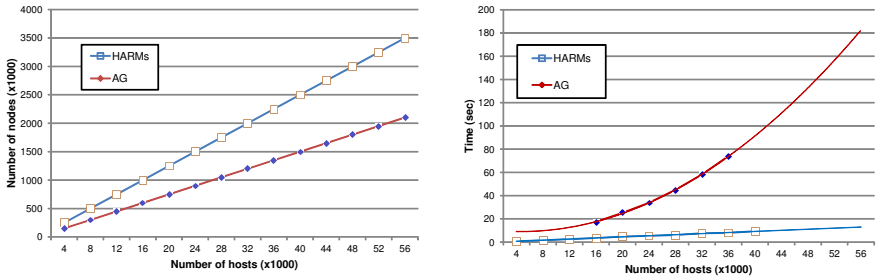


(a) Number of Edges (b) Construction Time Measurement

Fig. 2. A Comparison between AG and HARMs in the Construction Phase

attack models have linear growth of the edge numbers, but the number of edges for the HARMs was always less than that of the simplified AG.

The trend observed from the simulation is comparable with the simulation result of the MPG [11]. The time comparison shows that the time for the evaluation increases rapidly for the simplified AG, but almost linearly does for the HARMs as shown in Figure 3(b). In contrast, the number of nodes computed in the HARMs is much greater than that of the simplified AG. The simplified AG constructs the attack paths using vulnerability sequences only, but the HARMs also analyse the sequence of hosts. Therefore, we require extra space of memory to store the information.



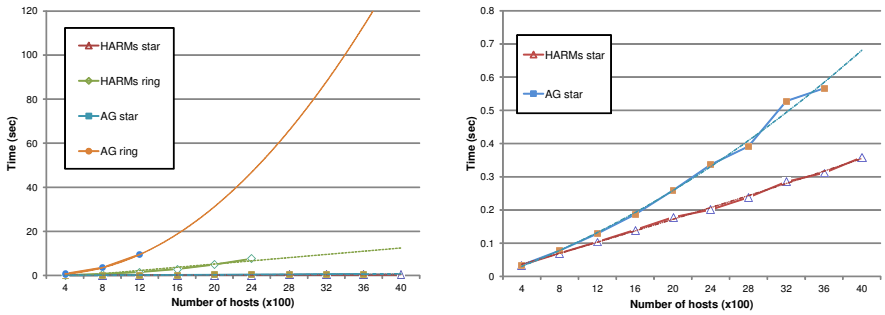
(a) Number of Nodes Computed (b) Evaluation Time Measurement

Fig. 3. A Comparison between AG and HARMs in the Evaluation Phase

4.2 Network Topologies and Vulnerabilities

We use various network topologies and variable number of vulnerabilities in our second simulation and compare it with the performance of the simplified AG. We incorporate bus, ring, and star topologies to connect hosts in each internal network, and

the number of vulnerabilities for each host is varied from 10 to 150. The number of hosts is fixed at 1200 when simulating the variable number of vulnerabilities. The same network structure was used, but the goal of the attack is to compromise a single host selected in the last subnet in the internal network (e.g., a host in the 10th subnet in each internal networks). The bridging hosts (i.e., head hosts that connect to other subnets) are not selected as the target host. In order to simulate different topologies, we assigned a single vulnerability to each host that is enough to gain the root access.



(a) Scalability of Different Network Topologies

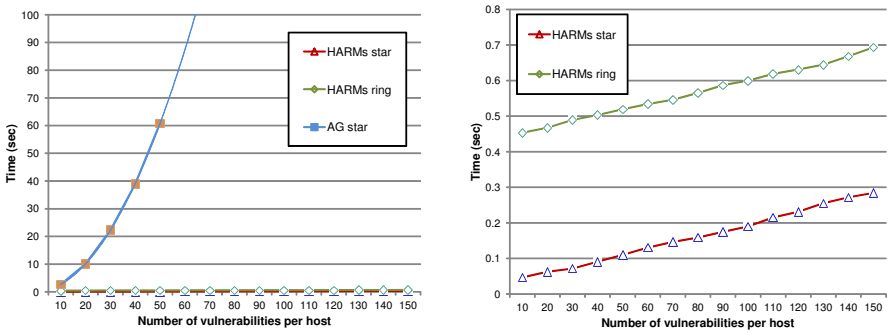
(b) Scalability of Star Topology

Fig. 4. Scalability Difference of Network Topologies in the Evaluation Phase

The simulation of different topologies is shown in Figure 4. Since the construction of the HARMs and the simplified AG is similar, we compare the different performance observed in the evaluation phase. The construction of the full path topology was computable, but the evaluation of the full path topology suffered from the scalability problem in the evaluation phase, where the evaluation of 400 hosts reached to the time out (i.e., it took longer than three hours). However, we observe that the simplified AG is slower than the HARMs when all topologies are taken into account significantly.

The simulation of varying the number of vulnerabilities is shown in Figure 5. The number of hosts was fixed at 1200. The fully connected topology for both attack models could not be evaluated for 1200 hosts. In addition, the ring topology for the simplified AG reached to the time out during the simulation (i.e., it took longer than three hours to evaluate). The comparison in the evaluation phase shows that as the number of vulnerabilities increase, the growth rate of the simplified AG is much greater than the HARMs for all network topologies. The trend for the simplified AG showed a quadratic increase, whereas the trend for the HARMs showed a linear increase in time. The slopes are almost linear for all topologies of HARMs, indicating the number of vulnerabilities is also a constant factor in the evaluation phase.

We simulated the performance of the simplified AG and the HARMs considering practical attack scenarios, various network topologies and variable number of vulnerabilities. Both attack models were built and analysed using the same networked system



(a) Evaluation Time (b) HARMs Star and Ring Topologies

Fig. 5. Scalability Difference with Varying Number of Vulnerabilities

in the simulation. The same method to construct the attack model was applied to both the simplified AG and the HARMs, and the result shows that the time measurement for the construction phase is similar. The same algorithm was used to compute the full attack paths, but we observe that the HARMs improved the performance in the evaluation phase dramatically.

5 Discussion

The efficiency of the HARMs is shown through comparisons with existing attack models and simulations. We listed some key questions to compare the scalability of attack models. The simulation shows that the scalability of the HARMs, and was compared with the simplified AG to show the efficiency of the HARMs in the construction and the evaluation phases. However, to improve the usability of the HARMs, we must consider the modification phase in case of update events in the networked system.

5.1 Scalability of attack model phases

The simplified AG suffered the scalability problem due to independent connections between the model components. The representation of the simplified AG had more edges compared with the HARMs. The number of nodes was the same, but the number of edges was greater in the simplified AG. However, the construction time shows that there is only a little difference between the HARMs and the simplified AG.

A few existing attack models compared the performance against the simplified AG in the construction and the evaluation phases. None of the attack models considered an update event in the networked system, and how their models are updated. The similarity between the simplified AG, LAG, and the MPG is that they are represented as a single layer in an attack model. Those attack models suffer from a scalability problem in the representation, and also the modification may affect all nodes in the attack model in

the worst case. However, hierarchical models, such as the HARMs and the TLAG, have less structural changes as they have less relationship between nodes than attack models that are represented in a single layer. In addition, the HARMs have fewer components to update, because the TLAG has higher number of lower layer models.

5.2 Network Structure and Attack Scenarios

There are many different network structures and types threatened by cyber attacks. The worst case complexity defines the upper bound performance for the HARMs. We have built two different attack scenarios to compare the scalability of the HARMs and the simplified AG through simulations. The first attack scenario covered a practical network structure. The attack scenario was simulated, and the result shows that the performances of both the HARMs and the simplified AG are much better than the defined complexities. However, the improvements observed are proportional to their theoretical complexities. Thus, the complexity measurements are good indicators to estimate the performance of the HARMs. The efficiency of the HARMs is shown in the evaluation phase, where the HARMs outperform the simplified AG. The simulation study showed a clear benefit of using the HARMs.

The second simulation compares the scalability of different network topologies, and how much the scalability is affected when the number of vulnerabilities increased. The results were comparable with some of the existing attack models and their analyses [5, 11]. The comparison between the HARMs and the simplified AG shows the performance of the HARMs was always better than the simplified AG. The quadratic growth trend of the simplified AG in the evaluation phase was comparable against the HARMs, where the growth trend was almost linear. However, we only considered a single network topology for each simulation. The networked system consists of different network topologies, but this is not modelled in our study. To accurately measure the expected performance of the networked system and its attack models, combinations of network topologies need to be modelled and simulated.

The variation of vulnerabilities affected the simplified AG significantly, showing an almost exponential growth in the evaluation phase. However, the HARMs showed a linear growth of the evaluation time, which is practically computable for a large number of vulnerabilities. Because the underlying algorithms are the same (e.g., construction algorithm, full path search algorithm), the improvement of scalability comes from the structural advantages of the HARMs.

The performance between the HARMs and the TLAG is not compared in the simulation because identical hosts were used. A network with homogeneous hosts will result in HARMs and TLAG having the same number of upper and the lower layer components. A further comparison is required using a network with heterogeneous hosts to distinguish the HARMs and the TLAG performances.

Since our focus was on comparing the scalability of current attack representation models, we have not considered a real system because we have assumed that the complexity in each host is linearly proportional to the number of hosts (i.e., a constant factor). However, the complexities in real systems are difficult to represent in a simulation, and various network protocols and services may affect how the network traffic

flows, such that considering the time in the security analysis may vary the result. We will consider a real system in our future works.

5.3 The Simplified AG and the HARMs

The simulation demonstrated the improvements of existing attack models using the same underlying attack model and algorithms. The time measurement for the construction phase was similar, but the simplified AG showed that it created more edges than the HARMs. Consequently, the performance of the evaluation phase shows that the evaluation time for the simplified AG has increased more rapidly compared with the HARMs, where the growth of the HARMs evaluation time was almost linear. The underlying algorithm to compute the attack paths was the same, but we observe that the performance of the HARMs is more efficient than the simplified AG. The structure of the HARMs reduces the total number of edges in the attack model, so we require fewer computations during the evaluation phase.

In the evaluation process, the number of nodes used in the computations was captured in the simulation. The simulation showed that the number of nodes in the HARMs is greater than the simplified AG, because the upper layer components of the HARMs are also evaluated. As a result, more memory space is required for the HARMs. However, if we allocate the memory space efficiently (e.g., by freeing spaces used by the lower layer calculations when finished), we can reduce the extra memory required by the HARMs. Also, if the lower layer information has been changed, only the lower layer calculations are affected. As a result, the complexity of the HARMs is not largely affected. In contrast, the evaluation time for the simplified AG may fluctuate depending on the changes in the lower layer. In addition, we observed that the number of nodes is only one of the factors that affect the time complexity in the evaluation phase. The clustering of nodes can reduce the time complexity dramatically, as shown in the simulation.

6 Conclusion

Attack models have evolved over the last decade to evaluate the network security. They can also provide countermeasures to enhance the security. The major hurdle of evaluating the network security considering the full attack paths is the scalability problem, where the number of possible attack scenarios grows exponentially as the number of hosts and vulnerabilities increase. Improvements to the full AG have been developed, and new types of attack models (e.g., LAG, MPG and TLAG) have been proposed to address the scalability problem. The HARMs are described and compared with some of the existing attack models to show the scalability improvements.

The efficiency of the HARMs is demonstrated through the simulation, where the underlying algorithms and models were the same to evaluate the full attack paths, but we observe that the performance of the HARMs was better than the simplified AG in the simulation. Moreover, the HARMs have better performance than computational complexities when a practical network scenario is considered.

References

1. Ammann, P., Wijesekera, D., Kaushik, S.: Scalable, graph-based network vulnerability analysis. In: Proc. of the 9th ACM conference on Computer and communications security (CCS 2002), New York, NY, USA, ACM (2002) 217–224
2. Dewri, R., Poolsappasit, N., Ray, I., Whitley, D.: Optimal security hardening using multi-objective optimization on attack tree models of networks. In: Proc. of ACM conference on Computer and communications security (CCS 2007), New York, NY, USA, ACM (2007) 204–213
3. Gupta, S., Winstead, J.: Using Attack Graphs to Design Systems. Security and Privacy, IEEE **5**(4) (2007) 80–83
4. Xie, A., Cai, Z., Tang, C., Hu, J., Chen, Z.: Evaluating network security with two-layer attack graphs. In: Proc. of Computer Security Applications Conference (ACSAC 2009). (2009)
5. Ou, X., Boyer, W., McQueen, M.: A scalable approach to attack graph generation. In: Proc. of the 13th ACM conference on Computer and communications security (CCS 2006), ACM (2006) 336–345
6. Ingols, K., Chu, M., Lippmann, R., Webster, S., Boyer, S.: Modeling modern network attacks and countermeasures using attack graphs. In: Proc. of Annual Computer Security Applications Conference (ACSAC 2009), IEEE (2009) 117–126
7. Saini, V., Duan, Q., Paruchuri, V.: Threat modeling using attack trees. J. Comput. Sci. Coll. **23**(4) (2008) 124–131
8. Dawkins, J., Hale, J.: A systematic approach to multi-stage network attack analysis. In: Proc. of Second IEEE International Information Assurance Workshop (IWIA 2004). (2004) 48 – 56
9. Edge, K.: A Framework for Analyzing and Mitigating the Vulnerabilities of Complex Systems via Attack and Protection Trees. PhD thesis, Air Force Institute of Technology (2007)
10. Sheyner, O., Haines, J., Jha, S., Lippmann, R., Wing, J.: Automated generation and analysis of attack graphs. Technical report, CMU (May, 2002)
11. Ingols, K., Lippmann, R., Piwowarski, K.: Practical attack graph generation for network defense. In: Proc. of Computer Security Applications Conference (ACSAC 2006). (2006) 121 –130
12. Lippmann, R., Ingols, K.: An Annotated Review of Past Papers on Attack Graphs. ESC-TR-2005-054 (2005)
13. Noel, S., Jajodia, S.: Managing attack graph complexity through visual hierarchical aggregation. In: Proc. of the 2004 ACM workshop on Visualization and data mining for computer security (VizSec 2004), ACM (2004) 109–118
14. Chen, F., Liu, D., Zhang, Y., Su, J.: A scalable approach to analyzing network security using compact attack graphs. Journal of Networks **5**(5) (2010)
15. Schneier, B.: Secrets and Lies: Digital Security in a Networked World. John Wiley and Sons Inc. (2000)
16. Roy, A., Kim, D., Trivedi, K.: Attack Countermeasure Trees (ACT): towards unifying the constructs of attack and defense trees. Security and Communication Networks **5**(8) (2012) 929–943
17. Hong, J., Kim, D.: HARMs: Hierarchical Attack Representation Models for Network Security Analysis. In: Proc. of the 10th Australian Information Security Management Conference in SECAU Security Congress (SECAU 2012). (2012)
18. Sawilla, R., Ou, X.: Identifying critical attack assets in dependency attack graphs. In: Proc. of the 13th European Symposium on Research in Computer Security (ESORICS 2008), Berlin, Heidelberg, Springer-Verlag (2008) 18–34

19. Noel, S., Jajodia, S.: Understanding complex network attack graphs through clustered adjacency matrices. In: Proc. of the 21st Annual Computer Security Applications Conference (ACSAC 2005). (2005) 10 pp. –169
20. Hewett, R., Kijsanayothin, P.: Host-centric model checking for network vulnerability analysis. In: Proc. Annual Computer Security Applications Conference (ACSAC 2008). (2008) 225 –234
21. Albanese, M., Jajodia, S., Noel, S.: Time-efficient and cost-effective network hardening using attack graphs. In: Proc. Dependable Systems and Networks (DSN 2012), Los Alamitos, CA, USA, IEEE Computer Society (2012)
22. Pawlikowski, K., Jeong, H., Lee, J.: On credibility of simulation studies of telecommunication networks. Communications Magazine, IEEE **40**(1) (2002) 132 –139
23. Ewing, G., Pawlikowski, K., McNickle, D.: Akaroa-2: Exploiting network computing by distributing stochastic simulation. In: Proc. European Simulation Multiconference (ISCS 1999). (1999) 175–181