DEFENSE GRAPHS AND ENTERPRISE ARCHITECTURE FOR INFORMATION ASSURANCE ANALYSIS

Ulrik Franke*, Teodor Sommestad, Mathias Ekstedt, and Pontus Johnson
Department of Industrial Information and Control Systems
Royal Institute of Technology (KTH)
100 44 Stockholm, Sweden

ABSTRACT

The JQRR metrics for Information Assurance (IA) and Computer Network Defense (CND) are combined with a framework based on defense graphs. This enables the use of architectural models for rational decision making, based on the mathematical rigor of extended influence diagrams. A sample abstract model is provided, along with a simple example of its usage to assess access control vulnerability.

1. INTRODUCTION

With the advent of Network Centric Warfare, Information Assurance (IA) is becoming ever more important to the success of military operations. Reliable and secure IT systems are vital to ensure success on the battlefield, and precisely because of this, they also become the focus of adversarial attention.

IA, however, is a complicated function of many different concepts such as technical countermeasures, organizational policies, security procedures, and more. Measuring the level of IA, therefore, is a non-trivial exercise; making rational decisions and prioritizations about the use of scarce resources is ever more so.

To efficiently protect computer networks and the information stored in them, combatant commanders and combat support agencies need to be able to assess the current security level of their IT systems as well as the security level after improvements. An example of a framework for such assessment is the Information Assurance (IA) and Computer Network Defense (CND) Joint Quarterly Readiness Review (JQRR) Metrics (Joint Chiefs of Staff, 2003), which provides six different categories of metrics, used for readiness assessments of US forces: 1. Personnel, 2. Training, 3. Operations, 4. Technology (equipment), 5. Supporting Infrastructure, and 6. Intelligence.

The diversity of these metrics, and similar ones, poses problems of how to accurately weigh them all together into a coherent picture of security. An even more pressing problem, however, is that all assessment metrics are *a priori*, while the actual threat consequences, of course, are *a posteriori* notions. This is *causal uncertainty*.

Furthermore, decision makers using metrics face a second kind of uncertainty, viz. whether information and indicators collected during a security assessment are credible. Measurement errors, misunderstandings and deliberate deception all challenge the credibility of the assessment result. This is *measurement uncertainty*.

This paper describes a method for how to combine Bayesian statistics-based extended influence diagrams with attack graphs and countermeasures into an IA assessment framework. This framework is able to take both types of uncertainty into consideration.

This approach allows a mathematical handling of the uncertainty regarding both what countermeasures are in place, and how well they contribute to thwarting attacks. The Bayesian approach allows calculating the probability that attacks succeed from an enterprise architecture model. The framework also takes uncertainties of the security assessment into consideration. Moreover, using the extended influence diagram formalism, the expected loss from each attack can be calculated. Scenarios can be compared, allowing more informed decisions of how to optimally use the available IA resources.

1.1 Outline

The remainder of this paper is structured as follows. Section 2 addresses some related works on security metrics and puts the present contribution into context. The important concepts of attack trees and defense graphs are introduced in section 3, whereas the extended influence diagrams used for probabilistic modeling are introduced in section 4. Section 5 provides a simple example of how to use the theory thus formed for IA analysis. Section 6 explains how the preceding theories can be integrated into a single abstract model. Section 7 summarizes the contribution, while section 8 concludes the paper.

2. SECURITY METRICS

Within the field of security and information assurance research, substantial efforts have been devoted to methods and methodologies to rank, score and measure security. In addition to direct metrics, such as "the percentage of systems updated with the latest patches", a number of more elaborate measurement and ranking methods have been suggested. Some examples are the weakest adversary metric (Pamula et al. 2006), mean-time-to-compromise metric (Leversage and James, 2008), robustness strategy (Arber et al., 2000), the attack surface metric (Manadhata and Wing, 2005), operational readiness metrics (Connolly, 2001), and the system vulnerability index (Alex-Foss and Barbosa, 1995).

IA depends on the interaction of processes, procedures, tools and people (Henning, 2001). One of the conclusions from the Workshop on Information-Security System Rating and Ranking (Henning, 2001) was that it will not be possible to successfully quantify the assurance present in a system using any one single security metric. Consequently, metric frameworks typically suggest multiple metrics for these different domains; see for example NIST SP 800-55 (Swanson et al., 2003) or the JQRR metrics (Joint Chiefs of Staff, 2003). Checklist methods based on standards such as ISO 17799 is another common practice to include the many facets of security. The methods typically provide a list of indicators, but do not describe how to combine these indicators into an overall value for security. Yet, an overall indicator on security is desirable, and methods to combine different metrics in a meaningful way is a subject of research. Some work has also been devoted to combing metrics into an overall indicator, for example (Weiss et al. 2005) and (Johansson, 2005).

However, no prior work has described how to combine metrics while taking into account both *causal uncertainty* and *measurement uncertainty*. This paper suggests a method for doing so by using attack trees as the structure for aggregating values related to security into a single measure.

3. ATTACK TREES AND DEFENSE GRAPHS

Attack trees are a graphical notation evolved from fault trees, where the main goal of an attacker is depicted as the root of a tree (Schechter, 2004). The steps to reach this goal are broken down into sub-goals of the attack through "AND" and "OR" relationships. This is a standard, intuitive way of modeling threats and security.

Attack graphs can easily grow extensively. To represent them more compactly, Liu and Hong (2005) have used Bayesian networks to express them and to calculate the probability of an attack against computer networks being successful based on vulnerabilities within it. These "Bayesian attack graphs" can be used to answer questions about the current security status and facilitate comparison with previous measurements, but does not answer questions about how to improve the security status. Bayesian networks have also been used together with attack trees to analyze other security related concepts, for example with the purpose of intrusion detection (Qin and Lee, 2004).

A natural extension of attack graphs is to include not only attacks, but also countermeasures. From the perspective of the system owner, this amounts to adding controllable elements to the graph. In (Howard and LeBlanc, 2003) countermeasures are modeled together with trees depicting threats, and in the theses by Foster (2002) and Schechter (2004) countermeasures are included in the tree structures. The concept of including countermeasures in the tree structure has also been used in (Bistarelli et al. 2006), to create something called "defense trees", illustrated in Figure 1. Techniques have been presented which use defense trees for strategic evaluation of security investments (Bistarelli et al. 2006), modeling strategic games in security (Bistarelli et al. 2007b) as well as modeling of conditional preference of defense techniques using conditional preference nets (Bistarelli et al. 2007a). Defense trees (or graphs) has also suggested together with extended influence diagrams for security assessments in Sommestad et al (2008) and Sommestad et al. (2009). This paper builds on that work and describes how defense trees can be connected with measurement frameworks to create an aggregate indicator on security.

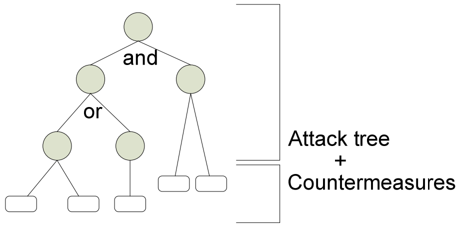
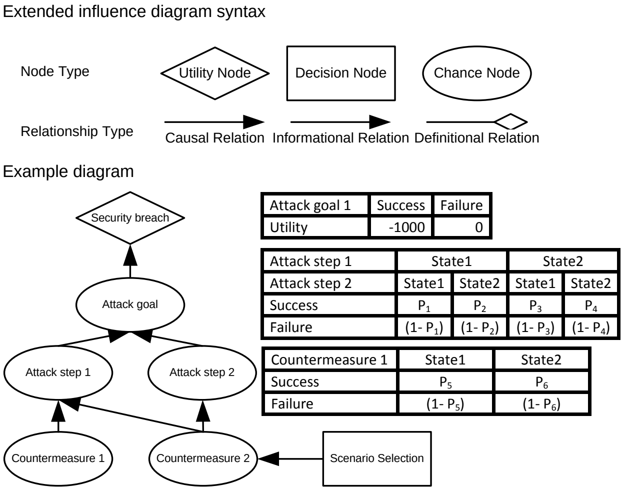


Figure 1. The defense tree concept, from (Bistarelli, 2007a).

4. EXTENDED INFLUENCE DIAGRAMS

Extended influence diagrams are a powerful modeling approach, used to depict and analyze complex causal interplay between quantities (Johnson et al., 2007b). These diagrams may be used to formally specify enterprise architecture analysis (Johnson et al., 2007a). The diagrams are an extension of influence diagrams, as described by Shachter (1986 and 1988) which in turn are an enhancement of Bayesian networks (cf. Neapolitan (2003) and Jensen (2001)). In extended influence diagrams, random variables graphically represented as chance nodes may assume values, or states, from a finite domain (cf. Fig. 2). A utility node represents a desired goal, such as "Information confidentiality". The meaning of the utility node can be further defined by other nodes that it has a definitional relation to. Causal relations on the other hand capture associations of the real world, such as “the training of system administrators affects network security”.

As illustrated in the example diagram of Fig. 2, Extended Influence Diagrams can be used to represent defense trees. A utility node can be used to represent the consequence of successful attacks and the steps required for their success can be decomposed into a number of substeps. Attack steps will then assume the state “Success” or “Failure”, depending on the states of its parents. The states of countermeasures influence the probability that an attack will be successful. Thus, they are modeled as causal parents to the attack steps. Finally, depending on the scenario chosen, the states of countermeasures will differ. This can be represented by decision nodes that influence the state of countermeasures. (Sommestad et al, 2008).



Attack goal 1	Success	Failure
Utility	-1000	0

Attack step 1	State1		State2	
Attack step 2	State1	State2	State1	State2
Success	P_1	P_2	P_3	P_4
Failure	$(1- P_1)$	$(1- P_2)$	$(1- P_3)$	$(1- P_4)$

Countermeasure 1	State1	State2
Success	P_5	P_6
Failure	$(1- P_5)$	$(1- P_6)$

Figure 2. Syntactic elements of extended influence diagrams and a simple example.

The mathematical rigor describing the causal relations is that of Bayesian networks. A Bayesian network, $B = (G, P)$, can be described as a representation of a joint probability distribution, where $G = (V, E)$ is a directed acyclic graph consisting of vertices, V, and edges, E. P is the probability distribution over the states of the variables associated with each vertex.

In a Bayesian network, the vertices denote a domain of random variables $X_1,\ldots,X_n$, also called chance nodes. In the context of concrete models, each chance node corresponds to an attribute. Each chance node, X_i, may assume a value x_i from the finite domain $Val(X_i)$. The advantage of the graph representation is that it provides a compact way of expressing the dependency relations between the random variables, i.e. which variables are conditionally independent given other variables. Each edge denotes a causal dependency between its nodes.

In order to specify the joint distribution, the respective conditional probabilities that appear in the product form (1) must be defined.

$$P(X_1,\ldots,X_n) = \prod_{i=1}^{n} P(X_i | Pa(X_i)) \qquad (1)$$

The second component P describes distributions for each possible value x_i of X_i, and $pa(X_i)$ of $Pa(X_i)$, where $Pa(X_i)$ is the set of parent nodes of X_i. These conditional probabilities are represented in matrices, here forth called Conditional Probability Matrices (CPMs). Using a Bayesian network, it is possible to answer questions such as what is the probability of variable X being in state x_1 given that its parents Y and Z are in states y_2 and z_1 ($Y = y_2$ and $Z = z_1$). An example of a Bayesian network with CPMs representing the probabilities of success in various attacks is shown in Figure 2.

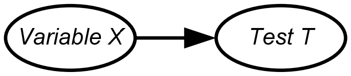
One important feature of the Bayesian formalism is the possibility to learn from previous data and create powerful statistical models for accurate IA assessments. Since extended influence diagrams, as opposed to mere Bayesian networks, include the notions of decision and utility nodes, predicted losses from successful attacks can be included in the models, thus enabling a more holistic view of IA.

4.1. Tests in Extended influence diagrams

In Bayesian networks and extended influence diagrams, entities are often modeled that are exceedingly difficult to assess directly. When modeling high level architectural concepts such as information assurance, system availability, etc. there is rarely a single gold standard of measurement. In the models, this is reflected by the use of *tests*. Tests can be done at different abstraction levels. At the lowest abstraction level, it is often straightforward to define and measure things like the percentage of computers that are fit with antivirus software. At a higher level, one might interview stakeholders about things such as the overall competence of system administrators, and skip the details of how they acquired this knowledge.

A common feature of such tests is that they do not reveal definite truths. Rather, test have a level of credibility that can be taken into account when performing the analysis.

Formally, a test of a variable is represented as a node, and the causality arrow is directed from the variable to the test. Thus, as it should be, the result of the test depends on the state of the variable, as illustrated in Figure 3. The states of the test, $\{t_1, t_2\}$, by definition correspond to the states $\{x_1, x_2\}$ of the variable as illustrated in the table in Figure 3.



X		x_1	x_2
T	t_1	$P(t_1 \mid x_1)$	$P(t_1 \mid x_2)$
	t_2	$P(t_2 \mid x_1)$	$P(t_2 \mid x_2)$

Figure 3. A node *X* with the test *T* and the CPM for the test node.

In the table in Figure 3, the outcome of the test is related to the actual states of the variable, i.e. a model of the accuracy of the test. A perfect test would correspond to an identity matrix CPM. Since most realistic (and interesting) tests are less than perfect, the CPM will rarely be an identity matrix, but rather reflect measurement uncertainty.

5. A SERVER ATTACK EXAMPLE

Figure 4 illustrates a defense tree for spoofing attacks directed against servers using the Internet Protocol (IP), inspired by (Howard and LeBlanc, 2003). Examples of military IP based networks include the NIPRNet and the SIPRNet, as well as the tactical voice over IP network RIPRNet. A spoofing attack will require the adversary to both knock out the valid machine and at the same time have created a new one with the same name. Knocking out a server can be done in four ways. Firstly, the attacker may hijack the DNS of the server by infecting it with malicious code or exploit some other vulnerability. Secondly, if the adversary is able to bypass the network perimeter with traffic without being cut off; she can flood the server with bad IP packets to make it unavailable. Two other options are simply to turn off the power to the server, or rename it into something else. Decomposing this further, the adversary is required to gain access to power breakers or to turn off the power, and require her access to the computer room and server interface to rename it, respectively.

A number of defense measures can mitigate this threat or at least make the steps in this attack more difficult to accomplish. In this simplified example, antivirus software and patched systems will provide some protection against attacks directed towards the DNS server. Having this functionality at the web server's local DNS server naturally does not offer protection against compromised servers at the client side. However, it does make attacks against the server side more difficult.

The ability to resist flooding of the server with bad IP packets is strengthened by using proper boundary protection (firewalls) and the use of intrusion detection systems (IDS) that can alert administrators of anomalies. Access to the computer room as well as its power can be restricted using physical access control mechanisms. Unauthorized access to the server can be mitigated with logical access controls, such as password protection.

This example illustrates attack vectors and countermeasures for an attack that spoofs a content server on an IP based network, such as the NIPRNet.

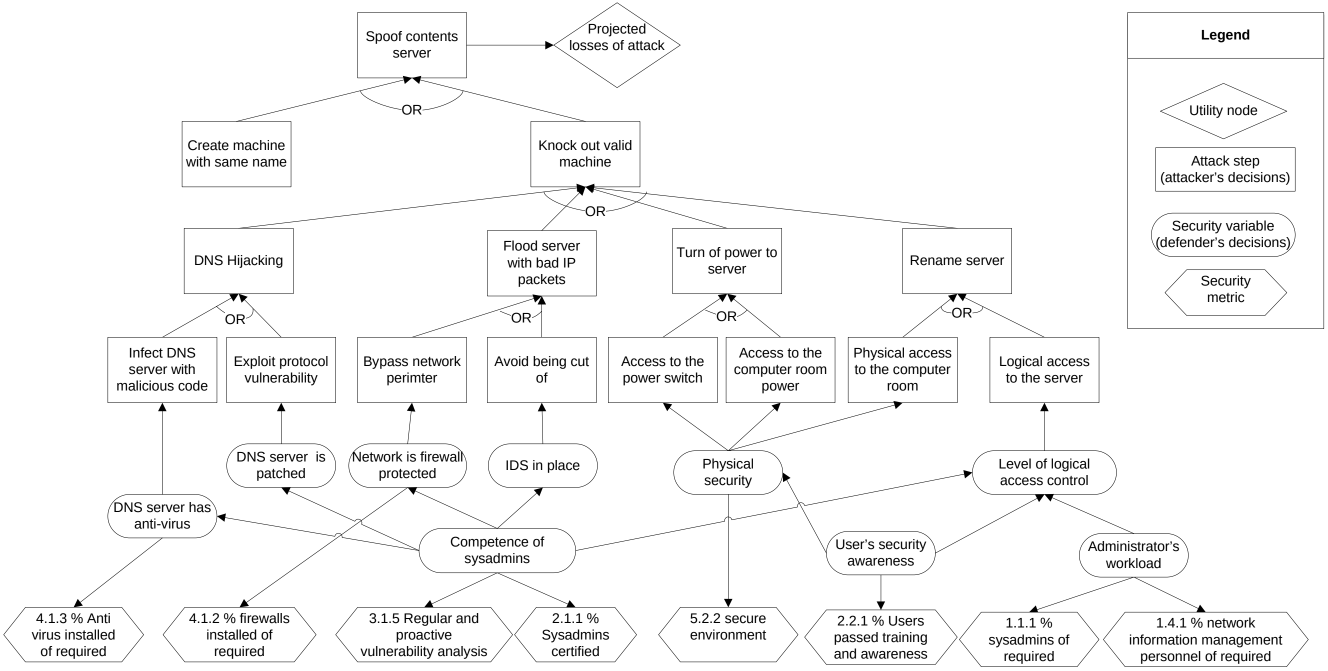

Spoof contents server
Projected losses of attack
Legend
Utility node
Attack step (attacker's decisions)
Security variable (defender's decisions)
Security metric
OR
Create machine with same name
Knock out valid machine
OR
DNS Hijacking
Flood server with bad IP packets
Turn of power to server
Rename server
OR
OR
OR
OR
Infect DNS server with malicious code
Exploit protocol vulnerability
Bypass network perimter
Avoid being cut of
Access to the power switch
Access to the computer room power
Physical access to the computer room
Logical access to the server
DNS server is patched
Network is firewall protected
IDS in place
Physical security
Level of logical access control
DNS server has anti-virus
Competence of sysadmins
User's security awareness
Administrator's workload
4.1.3 % Anti virus installed of required
4.1.2 % firewalls installed of required
3.1.5 Regular and proactive vulnerability analysis
2.1.1 % Sysadmins certified
5.2.2 secure environment
2.2.1 % Users passed training and awareness
1.1.1 % sysadmins of required
1.4.1 % network information management personnel of required

Figure 4. Example defense three. Inspired by (Howard and LeBlanc, 2003). The attack steps are the controllable nodes of the attacker, while the security variables are the controllable nodes of the system owner.

A security assessment investigating the possibilities of various adversarial attacks will assess the probability of success associated with the different attack paths in the model, as remedied by the associated security controls implemented. The JQRR metrics provides in total 82 metrics that are used to assess the level of IA and computer network defense (CND) of DoD information systems. Some of these are *a posteriori* indicators on the historical success rates of hostile attacks, while others indicate the current state of countermeasures. For clarity and brevity, a suitable subset of these metrics has been selected for the purpose of the present example.

5.1. Incorporating metrics into the models

As described in section 4.1, it is often exceedingly difficult to directly assess complex attributes such as the level of information assurance at a military facility. However, more low-level attributes of security facilities, such as the percentage of networks that have firewalls, or the percentage of system administrators that are properly certified can usually be found. Such data can be regarded as evidence on the state of the true variables. For example, consider the competence of the system administrators employed. The competence of a system administrator is a non-tangible, complex attribute, affected by a number of factors such as general experience, previous postings, formal education, certifications, ability to work under time pressure, etc. Clearly, these cannot all be modeled. What can be done, however, is an assessment based on a few simple attributes, such as the percentage of system administrators that are certified and whether regular and proactive vulnerability analyses are carried out.

Table 1. A sample CPM, relating the competence of system administrators to a measurable variable.

Competence of sysadmins		**High**	**Low**
Regular and proactive vulnerability analyses	**Yes**	0.95	0.50
	No	0.05	0.50

In Table 1, an example is given of how the competence of system administrators might be related to the existence of regular and proactive vulnerability analyses. It is reasonable to assume that such analyses occur with a very high probability if the administrators are highly competent, while less competent administrators will not be equally heedful.

Table 2. A sample CPM, relating the competence of system administrators to their level of certification.

Competence of sysadmins		**High**	**Low**
Percentage of certified sysadmins	**Yes**	0.80	0.30
	No	0.20	0.70

Similarly, Table 2 shows how the competence of system administrators might be related to their level of certification. Now, assuming that these fairly straightforward relations hold, inference about the abstract and more elusive competence of the system administrators can be carried out using Bayes' theorem.

5.2. The JQRR metrics

The JQRR metric 3.3.1 is an *a posteriori* indicator that deals with incidents of unauthorized access during the last reporting period. Put in the context of this example, this indicates whether the attack is possible or not and it would indicate whether access can be gained to the contents server or the DNS server.

Figure 4 further includes a number of metrics that indicate the state of countermeasures. The percentage of computers with antivirus software installed (JQRR 4.1.3) provides an indicator on whether the DNS server has such software installed. Metric 3.1.5, measures the readiness of regular and proactive vulnerability assessments and gives information that indicates whether systems are sufficiently patched and updated.

The percentage of firewalls that are installed as percentage of the number of required ones (JQRR 4.1.2) indicates whether the network is firewall protected. Metric 2.1.1 is the percentage of system administrators that are certified. This is assumed to be an indication of the quality of the network's firewall protection and IDS.

Physical security is assessed through JQRR metric 5.2.2. Logical access management is measured through three indicators: (1) The number of system administrators compared to the number required, (2) the percentage of users that has passed training and awareness requirements, and (3) the percentage of network information management personnel compared to the number required. These three jointly indicate the state of logical access control management.

6. ABSTRACT MODEL

The preceding example sketches but one out of many possible attacks and uses but a few of all the JQRR metrics. In order to generalize this example and employ the method proposed on a broader scale, it is necessary to approach the problem in a more abstract fashion. The concept of *metamodels* helps us to do that.

A metamodel is a collection of concepts, used as building blocks when modeling the world. A metamodel formalizes the fact that certain entities and relationships (e.g. "computer" and "firewall") are particularly important to include in IA assessment models. Having identified these concepts and relations, they can be used as templates in practical modeling cases.

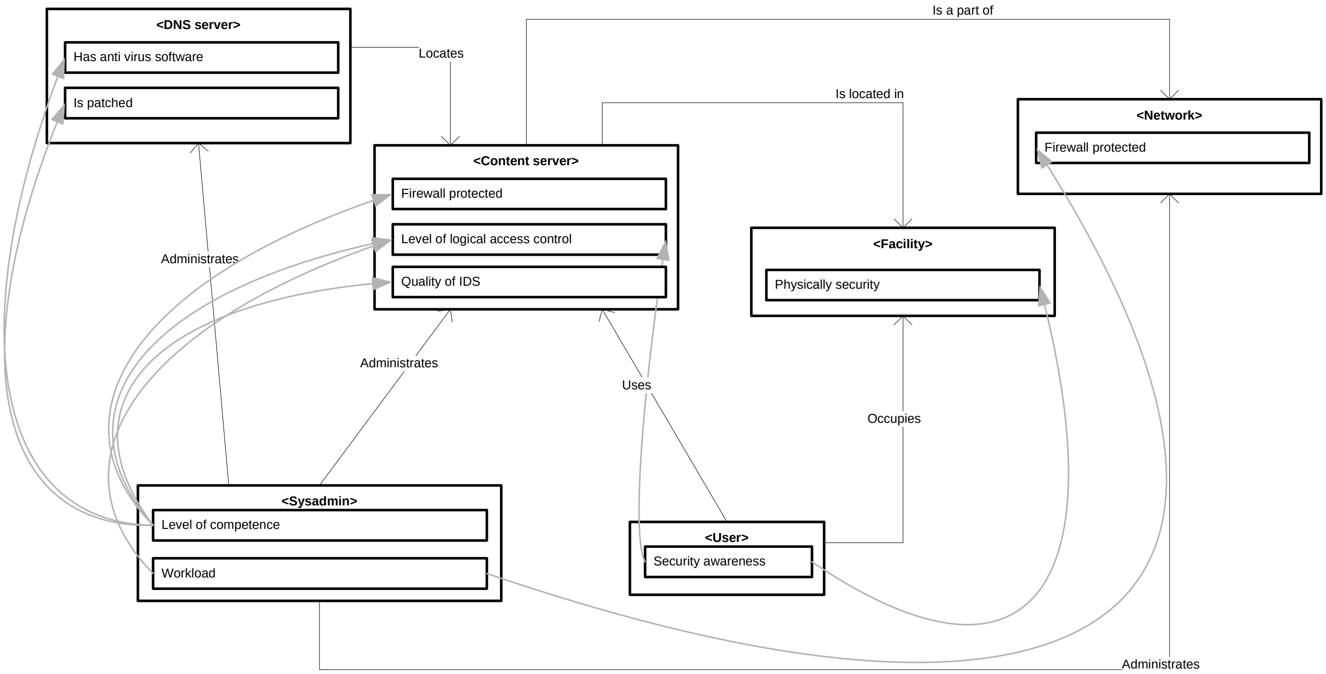


Figure 5. A metamodel describing the entities and relations used in the defense part of the attack example.

The use of metamodels guides modelers, enforces coherent practices and terminology, and enables the use of underlying theoretical concepts. Figure 5 illustrates a metamodel of the entities and relations used in the attack example depicted in Figure 4. As is readily seen, this metamodel matches the example and the domain of analysis in the sense that a modeler who uses its concepts as her building blocks will inevitably create a model well suited for IA analysis.

The usefulness of metamodels, however, is even more evident when considering their link to the mathematical formalism embedded in the extended influence diagrams. Each attribute relation corresponds to a probabilistic effect of one attribute on another, and to a CPM quantifying this. The term *abstract model* is used for a metamodel that is augmented with an extended influence diagram describing, in a Bayesian fashion, the causal relationships between the entities involved (Johnson et al, 2007b).

Detailed guidance for modeling of DoD systems and operations is beyond the scope of this paper, but can be found for instance in the DoD Architecture Framework (Department of Defense, 2007) and in the extensive related literature.

Using the metamodel depicted in Figure 5, concrete situations can be modeled and assessed with respect to IA. In Figure 6, a simple situation is described, using entities and relations from the metamodel. System administrators Kevin and James administrates servers (one DNS server and one NIPRNet contents server), and users Douglas, Robert and Kim all use either the contents server or have access to the building where it is located. By prescribing a terminology for describing this situation, the metamodel facilitates analysis of the concrete model using the extended influence diagram formalism. A concrete outcome of such an analysis might be a 32% risk of server spoofing, entailing an expected monetary loss of $ 2 million. Another outcome might be a recommendation on how to enhance the IA level.

7. DISCUSSION

The method described in the previous sections provides a framework for IA analysis with a number of notable strengths. Firstly, the use of abstract models integrates the use of existing metrics with the Bayesian formalism.

Secondly, the Bayesian formalism is well suited to handle both causal and measurement uncertainty, thus making the most of each IA assessment. Together with historical data on attacks, this facilitates calculation of expected loss for both the current state of systems and potential future scenarios.

Thirdly, information on expected losses prior to and after IA improvements enables more rational decision making. Using the framework proposed, combatant commanders and their staffs can create models of current and potential future scenarios based on metamodels covering the concepts relevant to IA.

It is worth to dwell on the possibility of training the underlying Bayesian network using historical data.

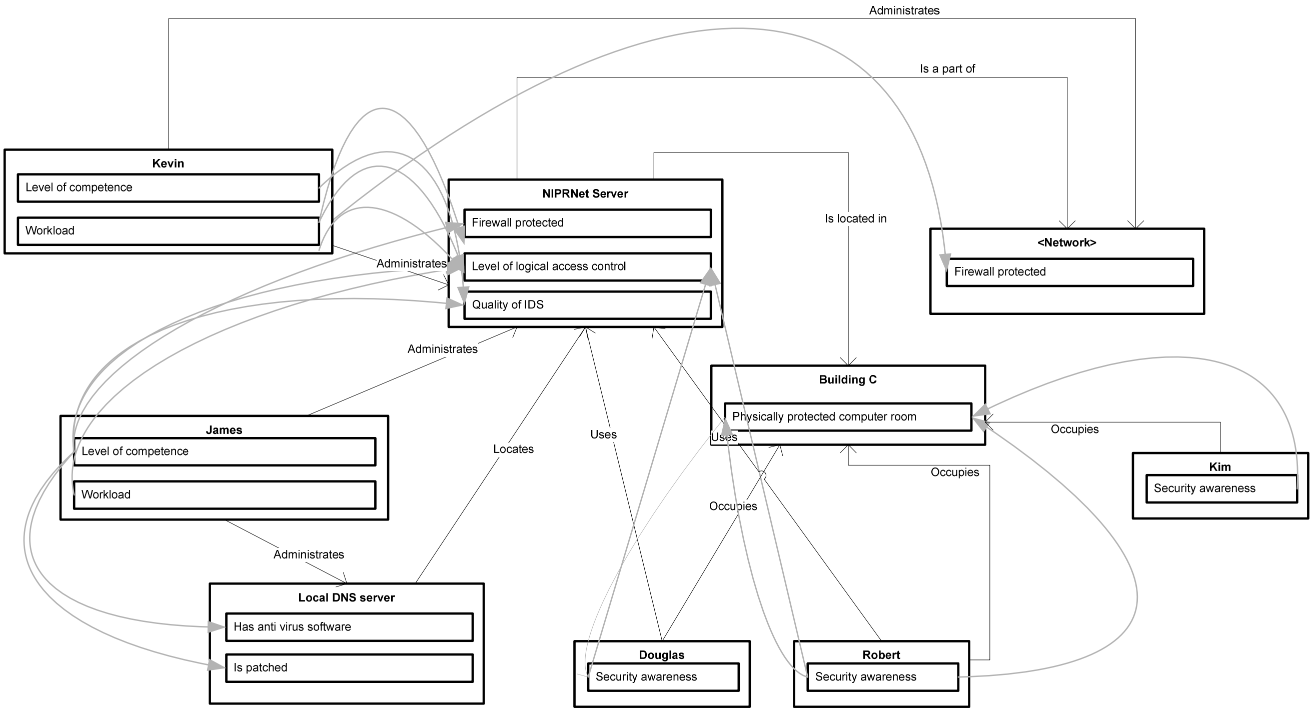


Figure 6. A concrete model describing entities and relations that could be involved in an actual IA scenario.

The present JQRR IA and CND metrics have been used by US defense services and DoD combat support agencies since July 2003. This means that a lot of data has been amassed and can be put to use within an extended influence diagram framework.

The training of Bayesian networks is a subject extensively treated in the literature (Jensen, 2001), (Russell and Norvig, 2003), (Friedman, 1998). Bayesian networks can be trained using expectation-maximization (EM) algorithms, which in simple cases essentially reduce to the standard Bayesian inference algorithm. More complicated cases can become computationally too complex and thus require methods such as Markov Chain Monte Carlo (MCMC) for approximate Bayesian learning. An example of a freely available software tool for EM learning of Bayesian networks is GeNIe, developed by the Decision Systems Laboratory at the University of Pittsburgh.

An extended influence diagram that has undergone proper learning with *a posteriori* measurement data becomes a powerful tool to assess the current IA level of military units and DoD combat support agencies. Furthermore, it provides a compact and intuitive representation of complex dependencies within the IA domain, leading to increased usability.

8. CONCLUSIONS

The present paper uses the JQRR IA and CND metrics, and shows how their use can be extended and improved within a framework based on defense graphs. A sample abstract model was provided, along with a simple example of its usage to assess access control vulnerability of an IP based military system such as the NIPRNet. The prospects for training a probabilistic inference engine based on historical data were discussed and identified as a potentially powerful method for making more rational IA assessments, a key factor in information warfare.

REFERENCES

Alex-Foss, J. and Barbosa, S. (1995): Assessing computer security vulnerability, ACM SIGOPS Operating Systems Review, Vol 29, Issue 3.

Arber, T., Cooley, D., Hirsch, S., Mahan, M., Osterritter, J., (2000): Network security framework: Robustness strategy, In Proceedings of 22nd National Information Systems Security Conference, Baltimore, MD.

Bistarelli, S, Dall'Aglio, M., Peretti, P, (2007a):"Strategic games on defense trees", Formal Aspects in Security and Trust, Springer Berlin / Heidelberg, , pp. 1-15.

Bistarelli, S., Fioravanti, F. and Peretti, P., (2006): Defense trees for economic evaluation of security investments, *Proceedings of Availability, Reliability and Security (ARES)*, 416 - 423.

Bistarelli, S., Fioravanti, F., Peretti, P., (2007b): Using CP-nets as a Guide for Countermeasure Selection, Proceedings of the 2007 ACM symposium on Applied computing, Seoul, Korea, pp. 300-304 .

Connolly, J. (2001): Information Assurance operational readiness metrics. In proceedings of the Workshop on Information-Security-System Rating and Ranking held in Williamsburg, VA, May 21-23.

Department of Defense (2007): DoD Architecture Framework version 1.5, volumes I, II, and III.

Foster, N. L. (2002): The application of software and safety engineering techniques to security protocol development. PhD thesis, Univ. of York, Department of Computer Science.

Friedman, N. (1998): The Bayesian Structural EM Algorithm. In Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence (UAI '98), San Francisco, CA.

Henning, R. (ed), (2002): Proceedings of the Workshop on Information Security System Scoring and Ranking (WISSRR) 2001 held in Williamsburg, VA, May 21-23.

Howard, M. and LeBlanc, D., (2003): Writing Secure Code, Second Edition, Microsoft Press, Redmond, Washington, USA.

Jensen, F.V., (2001): Bayesian Networks and Decision Graphs, Springer New York, Secaucus, NJ, USA.

Johansson, E. (2005): Assessment of Enterprise Information Security - How to make it Credible and Efficient, PhD Thesis, Royal Institute of Technology (KTH).

Johnson, P., Lagerström, R., and Närman, P, (2007a): Extended Influence Diagram Generation. Enterprise Interoperability II – New Challenges and Approaches, Springer London, 599-602.

Johnson, P., Lagerström, R., Närman, P., Simonsson, M. (2007b): Enterprise Architecture Analysis with Extended Influence Diagrams. *Inf Syst Front*, **9(2)**, Springer Netherlands, pp. 163-180.

Joint Chiefs of Staff, 2003 (2007): Chairman of The Joint Chiefs of Staff Instruction CJCSI 3401.03A: Information Assurance (IA) and Computer Network Defense (CND) Joint Quarterly Readiness Review (JQRR) Metrics, 15 July 2003, Current as of 10 July 2007.

Leversage, D.J. and James E., (2008): Estimating a System's Mean Time-to-Compromise, IEEE Security & Privacy, Volume 6, Issue 1, pp. 52-60.

Liu Y. and Hong, M., (2005): Network vulnerability assessment using Bayesian networks, Proceedings of Data Mining, Intrusion detection, Information assurance and Data networks security, Orlando, Florida, USA, pp 61-71.

Manadhata P. and J. M. Wing. An attack surface metric. In Technical Report CMU-CS-05-155, 2005.

Neapolitan, R (2003): Learning Bayesian Networks. Prentice-Hall, Inc. Upper Saddle River, NJ.

Pamula, J., Ammann, P., Jajodia, A. and Swarup V, (2006): A weakest-adversary security metric for network configuration security analysis, Conference on Computer and Communications Security, Proceedings of the 2nd ACM workshop on Quality of protection.

Qin, X. and Lee, W. (2004): Attack plan recognition and prediction using causal networks, in Proceedings of the 20th Annual Computer Security Applications Conference, Tucson, AZ, USA, pp. 370-379.

Russell, S. and Norvig, P (2003): Artificial Intelligence: A Modern Approach, Prentice-Hall, Inc. Upper Saddle River, NJ.

Schechter, S.E., (2004): Computer Security Strength & Risk: A Quantitative Approach. PhD thesis, Harvard University.

Shachter, R., (1988): Probabilistic inference and influence diagrams. Operations Research, 36(4), Hanover Maryland, pp. 36-40.

Shachter, R., (1986): Evaluating influence diagrams. Operations Research, 34(6), Institute for Operations Research and the Management Sciences, Hanover Maryland, pp. 871-882.

Sommestad, T., Ekstedt, M., Johnson, P., (2008): Combining defense graphs and enterprise architecture models for security analysis, Proceedings of the 12th IEEE International Enterprise Computing Conference.

Sommestad, T., Ekstedt, M., Johnson, P., (2009): Cyber Security Risks Assessment with Bayesian Defense Graphs and Architectural Model, Hawaii International Conference on System Sciences (HICSS), to appear.

Swanson, M., Bartol, N., Sabato, J., Hash, J. and Graffo, L. (2003): Security Metrics Guide for Information Technology Systems, NIST Special Publication 800-55.