Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 2012 Proceedings | Proceedings

Using Domain Knowledge to Facilitate Cyber Security Analysis

Peng He
Information Systems, UMBC, Baltimore, MD, United States., penghe1@umbc.edu

Lina Zhou
Information Systems, UMBC, Baltimore, MD, United States., zhoul@umbc.edu

George Karabatis
Information Systems, UMBC, Baltimore, MD, United States., georgek@umbc.edu

Using Domain Knowledge to Facilitate Cyber Security Analysis

Peng He
The Department of Information Systems
University of Maryland, Baltimore County
penghe1@umbc.edu

Lina Zhou
The Department of Information Systems
University of Maryland, Baltimore County
zhoul@umbc.edu

George Karabatis
The Department of Information Systems
University of Maryland, Baltimore County
georgek@umbc.edu

ABSTRACT

Network attack classification is essential to intrusion detection in that it could improve the performance of intrusion detection system. Several machine-learning methods have been applied to correlating attacks. However, the attack classification models developed in these methods share one common limitation in that they strongly rely on the training data, which can hardly be generalized to other datasets. To address the limitation, we propose to utilize domain knowledge in form of taxonomy and ontology to improve attack correlation in cyber security application. In addition, the attack correlations generated by machine-learning techniques are expected to refine the original attack taxonomy. We evaluate the proposed methods via a series of experiments. The findings of this study suggest that domain knowledge and machine-learning technique should be mutually beneficial and complementary to each other for attack classification.

Keywords

Taxonomy, Ontology, Cyber Security, Attack Correlation, Machine Learning.

INTRODUCTION

With the rapid growth of network applications nowadays, intrusion detection systems (IDS) have been widely deployed for monitoring network systems for malicious activities. In order to improve the detection rate of IDS, attack classification becomes an important task in intrusion detection. Several machine learning methods have been proposed and applied in correlating attacks [1]; however, these approaches are limited in that their results strongly rely on the training data. In addition, these methods alone do not perform well in intrusion detection because of their lack of knowledge from the cyber security domain [2, 3]. To address the limitation, we propose to utilize domain knowledge in form of Taxonomy and Ontology to improve attack correlation in cyber security. In addition, we integrate domain knowledge with machine-learning technique in attack classification.

In the rest of this paper, we start with a survey of existing taxonomies and ontologies in the cyber security domain, and illustrate their benefits for intrusion detection. Then, we discuss how we apply attack taxonomy to adjust attack correlations from semantic networks that are constructed based on the results of machine learning. Next, we demonstrate through an experiment that the attack correlations generated by machine learning techniques can be used to verify and refine the original attack taxonomy. Finally, we conclude the paper with findings and suggestions on attack classification.

RELATED WORK

In this section, we surveyed existing taxonomy and ontology in the cyber security domain and discuss their role and application in intrusion detection.

Taxonomies in Cyber Security Domain

A taxonomy is a classification system where the classification scheme conforms to a systematic arrangement into groups or categories according to established criteria. [4] According to Simpson [5], he defines a taxonomic character as a feature, attribute or characteristic that is divisible into at least two contrasting states and used for constructing classifications.

Taxonomies only provide schema for classification. They lack the necessary and sufficient constructs needed to enable a software system to reason over an instance of the taxonomy, which is representative of the domain under observation [6].

Several attack taxonomies have been proposed for the area of intrusion detection, which are summarized in table 1.Each of the taxonomies is discussed in detail next.

Taxonomy	Focused Aspects	Relations to other work
Bishop's vulnerability taxonomy	Vulnerabilities rather than attacks.	Earlier attack taxonomy
Howard's taxonomy	Processes.	Modification of Bishop's
Lough's taxonomy	The characteristics of attacks. Categories are mutually exclusive.	Refined Howard's to meet the "mutual exclusion" requirement.
Simon and Ray's taxonomy	A comprehensive scheme that contains four dimensions	References to Bishop's, Howard's and Lough's.
MIT Lincoln Lab's taxonomy	Attack consequence or goals	Inheritance from Simon and Ray's classification.

Table 1. A list of attack taxonomies and their contributions

Bishop's Vulnerability Taxonomy

In [7], Bishop presents a taxonomy of UNIX vulnerabilities in which the underlying vulnerabilities are used to create a classification scheme. He suggests that one of the main benefits of a taxonomy is that it should assist in the decision on resource investment. While this taxonomy focuses on vulnerabilities rather than attacks, as one of the most important early taxonomies in the cyber security field, it still provides a good background for proposing new taxonomies.

Howard's Taxonomy

Howard's approach in creating taxonomy of computer and network attack [8] is broad and process-based, which consists of five stages: attacks, tools, access, results and objectives. This taxonomy takes a process-driven rather than a classification perspective. However, Lough [9] points out that Howard fails to meet one of his taxonomy requirements – mutual exclusion, that taxonomy will categorize each attack into at most one category.

Lough's Taxonomy

In [9], Lough proposed a taxonomy based upon the characteristics of attacks, instead of using a tree-like taxonomy. The proposed four characteristics of attacks are the following: *improper validation, improper exposure, improper randomness and improper deallocation.* Simon [10] criticizes that Lough's taxonomy is general and does not discuss about attacks in terms of worms, viruses and Trojans, which is how attacks are usually described in practice, so it may not be useful for the day-to-day task of identifying and classifying new attacks.

Simon and Ray's Taxonomy

Simon [10] provides a common classification scheme that can be shared between organizations. Two general models for a taxonomy design are also discussed in their paper: tree-like structure or list based. The design of a tree-like structure will have more general categories at the top and specific categories at the leaves. However, how to deal with blended attacks in one tree-like structure is a problem; also, for attacks, they do not often have many common traits, thus it makes the creation of broad categories even harder. Another way to create taxonomy is by using lists. A flat-list with general categories can be considered or a flat-list with very specific categories can be proposed. Again, in order to deal with blended attacks, the categories may be too general, and consequently of limited use; or the categories are created so specific that the list will become too long with few instances within each category.

The taxonomy proposed in paper [10] took a different approach from either of the tree-like or flat-list taxonomies, despite that both of these approaches are used as components in their taxonomy. They proposed *four* major dimensions for attack classification. The *first*, which is also the base dimension, is used to categorize the attack into an attack class based on the

attack vector. The attack vector of an attack is the main means by which the attack reaches its target. The *second* dimension covers the targets of the attack, while the *third* dimension categorizes what is being used to attack the target; in other words, the *third* dimension covers the vulnerabilities and exploits that the attack uses, refers to a Common Vulnerabilities and Exposures (CVE) entry [11]. The last *four* dimension deals with attacks having payloads or effect beyond themselves. In addition, other dimensions are also proposed especially in regards to how to react to a new attack that falls into a certain category. For example, these added dimensions are: *Damage*, which attempts to measure the amount of damage that attack does; *Cost*, which for the money by cleaning up after an attack; *Propagation*, which applies more to replicating attacks; *Defense*, the methods by which an attack has been defended against could be made into a further defense dimension.

Table 2 below shows the results of classifying some attacks using Simon and Ray's four-dimensional taxonomy.

Attack	1st Dimension	2nd Dimension	3rd Dimension	4th Dimension
Blaster	Network-aware worm	MS Windows NT 4.0, 2000, XP, Server 2003	CAN-2003-0352 (CVE entry [10])	TCP packet flooding DoS
Use of John the Ripper	Guessing password attack	Unix family, Windows NT, 2000 & XP	Configuration (general type of vulnerabilities)	Disclosure of information
Sobig.F	Mass-mailing worm	Email client	Configuration (general type of vulnerabilities)	Trojan
Morris worm	Network-aware worm	BSD 4 Sun 3 & VAX variants	Implementation & Design (general type of vulnerabilities)	Theft of service & subversion

Table 2. Classification sample results

Simon and Ray [10] also use **a case study** to illustrate the classification process of the attack "Morris Worm", which is a blended attack consisted of a number of components including the Sendmail attack, the Fingerd attack and the Rsh/Rexec attack. In its *first dimension*, since the worm uses these three components to spread and uses network services to spread, therefore, the blended attack category is network-aware worm. In the *second dimension*, the worm attackes Sun Microsystems Sun 3 and VAX computers running BSD 4 variants, and the three above-mentioned components use the vulnerabilities to attack VAX and Sun 3 BSD variants. Given that the worm uses a number of vulnerabilities to spread, the broader categories are identified in the *third dimension* - Implementation & Design. Finally, the *fourth dimension* consists of two entries: theft of service (as the worm stole both network and computer resources) and subversion (as infected systems were used to propagate the worm).

Several benefits of using proposed taxonomy are discussed in the paper. The taxonomy provides a holistic approach to classifying attacks, which provides a useful and consistent taxonomy across different organizations. Moreover, the issues of how the blended attacks should be analyzed and described have been solved in a promising way. Authors also suggest utilizing their proposed taxonomy on the research of correlating attacks that previously may have appeared to have nothing in common but they can be related through one of these dimensions. They pointed out in the further work to move their taxonomy towards a knowledge base, which could detect correlation and assist in the process of classification implemented by artificial intelligence (AI).

MIT Lincoln Lab's Taxonomy

Weber et al. [12, 13, and 14] provided a taxonomy that defines the category by attack consequence or goals, during the 1998 and 1999 DARPA off line intrusion detection system evaluation. They include the sub-categories of *Denial of Service*, *Remote to Local*, *User to Root* and *Probe*. Example of the consequence categories are shown in table 3, which will be referred, in our research.

Attack Category	Attack Name
Denial of Service(DoS)	smurf, neptune, back, teardrop, pod, land
Remote to Local(R2L)	warezclient, guess_passwd, warezmaster, imap, ftp_write, multihop, phf, spy
User to Root(U2R)	buffer_overflow, rootkit, loadmodule, perl
Probe	satan, ipsweep, portsweep, nmap

Table 3. A taxonomy that defines the attack category by consequence

Each attack category is described as the following:

(1) *Denial of Service (DoS)*: It is designed to disrupt a host or network service, and the attacker make some computing or memory resource too busy to handle legitimate requests, or denies legitimate users access to a machine.

(2) *Remote to Local (R2L)*: It occurs when an attacker who has the ability to send packets to a machine over a network, but who does not have an account on that machine—exploits some vulnerability to gain local access as a user of that machine.

(3) *User to Root (U2R)*: It happens when a local user on a machine is able to obtain privileges to gain root access to the system.
(4) Probe: also named as "scan attacks", these include many programs that can automatically scan a network of computers to gather information or find known vulnerabilities. They can provide a map of machines and services and pinpoint weak points in a network, and thus, are more dangerous.

Undercoffer et al. [5] have incorporated these classifications into their work. By referring to their IDS ontology, the class attack has the property of "resulting in" and this construction is predicated upon the notion that an attack results in some consequence. The class Consequence is comprised of several subclasses that include *Denial of Service*, *Remote to Local*, *User to Root* and *Probe*, that are shown in figure 1.

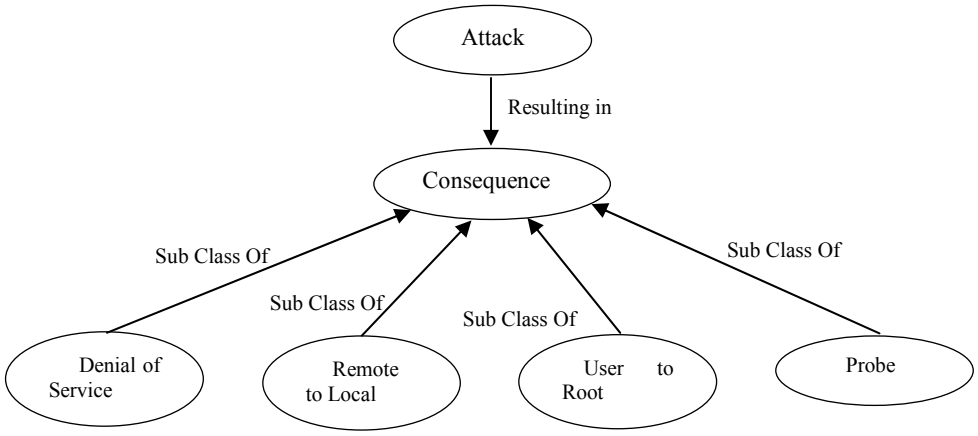

Figure 1. Part of IDS Ontology in paper [5]

Goals of a taxonomy have been discussed in several research papers. Weber [15] summarizes the desirable features for a general taxonomy of computer intrusions as the following: it can be used to guide evaluation of security tools, particularly IDSs; it can help prevent attacks in the future; it can be used to explain the cause of an attack. The criteria on how to evaluate a taxonomy is also discussed in them thesis.

Ontologies In Cyber Security Domain

Ontology, unlike taxonomies, provides more constructs that include machine interpretable definitions of the concepts within a specific domain and the relations between them. Gruber [16] defines an ontology as an explicit specification of a conceptualization. The term, which is borrowed from philosophy, is used to provide a formal specification of the concepts

and relationships that can exist between entities within a domain. Accordingly, ontologies are designed for enabling knowledge sharing and reuse between the entities within a domain.

According to Staab and Maedche [17], taxonomies differ from ontologies in that the former does not contain the necessary meta-knowledge required to convey modeling primitives such as concepts, relations and axioms that are required to make sense of and operate on specific objects.

There are several ways to build an ontology for a specific domain [18]. For instance, we can build a related ontology based on one taxonomy available in that domain. Moreover, a complete and well-formed ontology should include a taxonomy. On the other hand, we can reuse old ontology, and rebuild them based on up-to-date domain knowledge.

Compared with taxonomies, there are much less published research in creating ontologies in the intrusion detection area. The following are only two that can be found from the literature.

Target-Centric Computer Attack Ontology

The very first research by using semantic web methodology in intrusion detection area was done by Jeffrey Undercoffer et al. [5]. They suggested transitioning from taxonomies to ontologies in order to mitigate the effects of some problems caused by taxonomies, and they created an ontology specifying a model of computer attacks. To construct their ontology, they conducted an empirical analysis [19] of the features and attributes of over 4,000 classes of computer attacks and intrusions. They also analyzed their corresponding attack strategies and how they are categorized according to system component targeted means of attacks, consequence of attacks and location of attackers. The authors argued that any taxonomic characteristics used to define a computer attack are limited in scope to the features that are observable and measurable at the target of the attack, and they presented their model as a target-centric ontology.

They also illustrated the benefits of utilizing an ontology with use case scenarios. One of the scenarios is how to use pre-defined ontology to predict specific attack. DAML + OIL notation has been used to create instances of their ontology, and the representation of an instance of a neptune attack is illustrated in Figure 2 below. The first statement indicates that an event identified as 00035 has occurred and it has the *resulting _in* property, which is an instance of neptune identified with event number 00038.

```
<Intrusion:Host rdf:about="&IntrOnt;00035"
Intrusion:IP_Address="130.85.112.231"
rdfs:label="00035">
<Intrusion:resulting_in rdf:resource=
"&IntrOnt;00038"/>
</Intrusion:Host>
<Intrusion:nepture rdf:about="&IntrOnt;00038"
Intrusion:Exceed_T="true"
Intrusion:time="15:43:12"
Intrusion:date="02/22/2003"
rdfs:label="00038"/>
```

Figure 2. DAML+ OIL Notation for An Instance Of A Nepture Attack

When the ontology was queried for one of instances of Denial of Service (DoS) category attacks, the following statements are returned by the system shown in figure 3:

```
The event number of the intrusion is:
http://security.umbc.edu/Intrusion#00038
The type of intrusion is:
http://security.umbc.edu/Intrusion#neptune
The victim's IP address is:
130.85.112.231
The time and date of the event:
15:43:12 hours on 02/22/2003
```

Figure 3. Ontology querying results by the request of DoS

The authors pointed out in this use case that they only queried for the existence of a Denial of Service attack, but did not specifically ask for the type of nepture attack. However, the instance of the nepture attack was returned because it is a subclass of Denial of Service (DoS), which has been pre-defined in the ontology.

Attack Ontology in Distributed Intrusion Detection System

Another paper [20] also discussed about utilizing methods and techniques of semantic web in the Intrusion Detection Systems. They recognize that many of Intrusion Detection systems depend only on one kind of sources: network data or host data, and they do not have sufficient audit data, and thus they could cause high false positive and false negative. Their focus is on applying ontology in distributed intrusion detection environments and how to design these distributed systems to communicate with other systems.

In order to adapt to the distributed intrusion detection environments, their design contains different systems, each of which can have their own Intrusion Detection System, depends on the requirements, so they can do their detection individually. In addition, there is a central system that contains their proposed attack ontology. The author proposed to build their ontology based on Simon's taxonomy [10], which has four dimensions discussed in table 1, and the central system can extract the semantic relations among computer attacks, intrusions and suspect activities that occur in different systems in the network with this ontology.

There are different types of Intrusion Detection Systems, such as those are based on signatures, anomalies, network monitoring, host monitoring, etc. The key problem for designing one central system is how to correlate different kind of information from such distributed systems and report them to the central system. In their research, they modified a standard message format that can be converted into the format which is understandable for central system, for sending message via network to communicate between individual systems and central system.

The benefit of applying such an ontology is that it can solve the single point of failure problem in a network system. The single point of failure is a situation that if an intruder can prevent a central analyzer from working, for example, by crashing the host, the whole network would be unprotected. In their proposed distributed intrusion detection systems, every system has its own intrusion detection system. In other words, every system in the network utilizes two intrusion detection systems: the first is of its own and the second one is the ontology-based central detection system. The ontology in central system is also updated by other individual detection systems for attack properties from time to time in order to tackle most up-to-date attacks in the network.

PROPOSED METHODS

We propose that the domain knowledge and machine-learning techniques should be used together on attack classification task, which is expected to improve the performance of attack classification. The approach consists of two complementary components. On the one hand, we apply attack taxonomy in adjusting attack correlations from semantic networks that are based on machine learning techniques – similarity coefficients. On the other hand, we use the attack correlation results generated by machine learning techniques to verify and refine the original attack taxonomy. These two approaches are described in detail in this section.

Apply Attack Taxonomy In Adjusting Attack Correlations

We first proposed to utilize attack taxonomy as domain knowledge to adjust the attack correlation results generated by machine learning technique. In our previous research of applying semantic network in identifying network attacks [21], we created semantic networks for cyber security domain - each node represents an attack and the edges can be used to identify other relevant attacks according to one adjustable user-defined threshold for "relevance score". There were two steps involved in creating Semantic Networks: 1) automatically create First-Mode Semantic Network using similarity coefficients, such as Anderberg and Jaccard coefficient; and 2) extract domain expertise to fine-tune the previous attack correlation results as Second-Mode Semantic Network. In this paper, we mainly focus on how to apply domain knowledge to adjust the first-mode similarity-based semantic network.

Our experiment results with dataset such as DARPA intrusion detection evaluation data [22] show that solely applying similarity measures with the first-mode semantic network did not result in good classification results; nonetheless, it can serve as a good foundation for the further adjustment in correlating attacks. Moreover, external domain knowledge can be incorporated into adjusting similarity-based semantic network. Thus, we proposed the second mode semantic network that utilizes a mature attack taxonomy in modifying our first mode of similarity-based semantic network. We proposed that additional semantic information described as *semantic rules* can be extracted from domain knowledge represented as

Taxonomy and Ontology. Specifically, in this research, we adopted both the taxonomy from MIT Lincoln Lab and *semantic rule* as defined below:

Definitions [Semantic Rule]

*Given two attack nodes x_p and x_q, a semantic rule "**s**" is defined as: $\mathbf{s_{pq}}$ is the semantic score associating with these two attacks x_p, x_q based on domain knowledge such as Taxonomy. If x_p and x_q fall into same category of network attacks according to the Taxonomy, we set the value of $\mathbf{s_{pq}}$ to 1; otherwise, $\mathbf{s_{pq}}$ is set to 0.*

Then, $\mathbf{s_{pq}}$ is used to adjust the previous relevance score $\mathbf{rs_{pq}}$ in reference to a predefined threshold σ by domain experts. If $|\mathbf{s_{pq}} - \mathbf{rs_{pq}}| > \sigma$, then $\mathbf{rs_{pq}}$ needs to be updated by using semantic rules, and the degree of the adjustment Δ will also be customized by domain experts to reflect the relevance score among attacks in the current network environment. Hence, there are two possible ways to update previous relevance score:

(a) If $(\mathbf{s_{pq}} - \mathbf{rs_{pq}}) > 0$, the updated relevance score $\mathbf{RS'_{pq}} = \mathbf{rs_{pq}} + |\mathbf{s_{pq}} - \mathbf{rs_{pq}}| \times \Delta$

(b) If $(\mathbf{s_{pq}} - \mathbf{rs_{pq}}) < 0$, the updated relevance score $\mathbf{RS'_{pq}} = \mathbf{rs_{pq}} - |\mathbf{s_{pq}} - \mathbf{rs_{pq}}| \times \Delta$

Next, we illustrate how our second mode semantic network can be used to identify relevant attacks in the current network situation. A second mode semantic network is shown in figure 4 and the scores on the edges have been adjusted by *semantic rules* extracted from the attack taxonomy (See table 3):

ID	Actual Label	Bayesian Prediction	Relevant Attacks	Relevance Score
73727	**Perl**	Rootkit	**Perl**	**0.903333**
			warezclient	0.265921
			back	0.263353
			multihop	0.256667

Table 4. A concrete example of Semantic Network

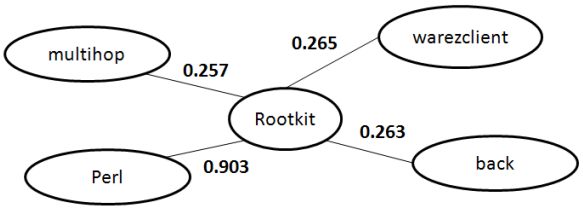

Figure 4. Example of second mode semantic network

In table 4, for the specific sequence of event (ID = 73727), Bayesian prediction model is first used to identify the attack type "Rootkit" as the highest probability of occurrence in the network. We then locate "Rootkit" as the initial node in our second mode semantic network, as shown in Figure 4. In addition, our approach will automatically indicate other attack types relevant to "Rootkit", and compute their relevance scores. The second column "Actual attack label" is used to validate the results of the semantic network. For instance, the attack "Perl" is predicted by the semantic network with the highest relevance score, and it happens to be the correct attack label for sequence ID = 73727. Thus, in this case, after utilizing attack taxonomy on the first mode similarity based semantic network, our second mode semantic network accurately detects probable attacks.

Verify and Adjust Attack Taxonomy Using Attack Correlations

The attack correlation results by similarity-based semantic network can also be used to validate and refine the original attack taxonomy, especially when such general attack taxonomy is applied in specific classification situation. This point is illustrated with a specific scenario in the following.

In our research, we first captured the attack signatures and created binary feature vectors, and then used four similarity measures in generating four different kinds of semantic networks based on these features, including Jaccard similarity coefficient, Anderberg similarity coefficient, Simple Matching similarity coefficient and Pearson's correlation coefficient. In order to compare the attack correlation results generated by our similarity-based semantic networks with the one described in the attack taxonomy, the same data source - DARPA intrusion detection evaluation dataset [22] was reused, and the attack taxonomy was adopted from MIT Lincoln Lab.

The experiment proceeded in the following steps: We first applied K-means clustering methods on the results of similarity-based semantic networks, and set K to four since it is expected to have four clusters according to the four categories in attack taxonomy from MIT Lincoln Lab. The analyses were performed by using K-means function from the Matlab software. Next, we analyzed these attack types from each of these clusters against the corresponding attack category as described in taxonomy, where the mismatched attacks were highlighted in grey and correct attack category labels were also marked. The analysis process has been repeated for four different semantic networks, as discussed above, and our preliminary results are reported in tables 5-8:

JC Kmeans (K=4)	Predicted Attack Name	Actual Attack Category based on the Taxonomy
1	portsweep satan guess_passwd	**Probe**
2	teardrop smurf spy pod ipsweep	**DoS**
3	perl ftp_write buffer_overflow loadmodule rootkit back multihop warezclient phf warezmaster	**R2L** + **U2R**
4	neptune imap nmap	**DoS**

	land	

Table 5. Clustering Results by Jaccard Similarity Based Semantic Network

AD Kmeans (K=4)	Predicted Attack Name	Actual Attack Category Based on the Taxonomy
1	neptune imap nmap land	**DoS**
2	teardrop smurf spy pod ipsweep	**DoS**
3	perl ftp_write buffer_overflow loadmodule rootkit back multihop warezclient phf warezmaster	**R2L** + **U2R**
4	portsweep satan guess_passwd	**Probe**

Table 6. Clustering Results By Anderberg Similarity Based Semantic Network

SMC Kmeans (K=4)	Predicted Attack Name	Actual Attack Category based on the Taxonomy
1	neptune land	**DoS**
2	portsweep satan guess_passwd	**Probe**
3	perl	**R2L**

	ftp_write buffer_overflow loadmodule rootkit back multihop warezclient phf warezmaster	+ **U2R**
4	teardrop smurf imap spy nmap pod ipsweep	**DoS** + **R2L** + **Probe**

Table 7. Clustering Results By Simple Matching Similarity Based Semantic Network

Correlation Kmeans (K=4)	**Predicted Attack Name**	**Actual Attack Category based on the Taxonomy**
1	portsweep neptune land satan guess_passwd	**Probe** + **DoS**
2	teardrop smurf imap nmap pod ipsweep	**DoS** + **Probe**
3	ftp_write buffer_overflow loadmodule rootkit back	**R2L** + **U2R**

	multihop warezclient warezmaster	
4	perl spy phf	**R2L**

Table 8. Clustering Results By Correlation Similarity Based Semantic Network

The experiment results show that the clustering results based on Jaccard and Anderberg similarity metrics are better than Simple Matching and Correlation similarity metrics. This is because the former metrics led to more homogeneous clusters in reference to the attack taxonomy of MIT Lincoln Lab, as shown in table 3.

Nonetheless, it is noted from the clustering results of Jaccard and Anderberg metrics, that there is still one combined cluster, which is supposed to be split into two separate clusters – R2L and U2R, according to the taxonomy as shown in table 3. The same problem occurs to the other two clustering results. Therefore, from the machine learning perspective, we speculate that the datasets in the categories of Jaccard and Anderberg have similar attack signature values, thus they are inclined to form into homogeneous clusters. However, according to the taxonomy in table 3, R2L and U2R are two separate attack categories. Thus, we speculate that only applying attack taxonomy may not be sufficient for such attack correlation task. Instead, machine learning and data mining techniques applied to original datasets should also be taken into consideration. That being said, after studying the original descriptions of two attack categories - from the taxonomy, we found that R2L and U2R do have similar behaviors in terms of gaining some kind of access – for examples, R2L occurs when attackers exploit some vulnerability to gain local access; and U2R takes place when attackers obtain privileges to gain root access.

CONCLUSION

In this paper, we surveyed existing taxonomies and ontologies in the cyber security domain, and illustrated their roles and applications in intrusion detection systems. In addition, we proposed a method that leverages one of attack taxonomies in improving attack correlations in semantic networks that are generated by machine learning techniques. Further, we demonstrate that the attack correlation results of machine learning technique, such as our similarity-based semantic network could also be used to verify and refine the original attack taxonomy, especially for specific classification situation. This study shows that attack taxonomy and machine learning techniques are complementary to each other and thus should be used together to improve the attack correlation results.

Only one taxonomy was used in the present study to evaluate the proposed method. In future work, we plan on integrating different attack taxonomies and ontologies for attack classification and applying such domain knowledge to other tasks involved in intrusion detection. Also, we are interested in doing some investigations to link the vulnerability to a specific user behavior, in order to show how the IDS data relates to the human behaviors factors; and we expect that they can also be captured in our taxonomies as domain knowledge to further assist attack correlation tasks.

ACKNOWLEDGMENTS

This research is partially supported by Northrop-Grumman Corporation.

REFERENCES

1. Reza Sadoddin and Ali Ghorbani. (2006) Alert correlation survey: framework and techniques, Proceedings of the 2006 International Conference on Privacy, Security and Trust: Bridge the Gap Between PST Technologies and Business Services.
2. Marco Barreno, Blaine Nelson, Russell Sears, Anthony D. Joseph and J. D. Tygar. (2006) Can Machine Learning Be Secure? *Proceedings of the 2006 ACM Symposium on Information, computer and communications security.*
3. Ling Huang, Anthony D. Joseph, Blaine Nelson, Benjamin I.P. Rubinstein and J. D. Tygar. (2011) Adversarial machine learning, *Proceedings of the 4th ACM workshop on Security and artificial intelligence, in AISec '11.*
4. WEBSTERS, editor. (1993) Merriam-Webster's Collegiate Dictionary. *Merriam-Webster, Inc., tenth edition*.
5. J Undercoffer, A. Joshi and 1. Pinkston. (2003) Modeling computer attacks: An ontology for intrusion detection, i*n: Springer. LNCS 2820 recent advances in intrusion detection.* Pittsburgh, pp 113- 135.
6. G. G. Simpson. (1961) Principals of Animal Taxonomy. *Columbia University Press*.
7. Bishop M. (1995) A taxonomy of (Unix) system and network vulnerabilities. *Technical Report CSE-9510*, Department of Computer Science, University of California at Davis.
8. Howard JD. (1997) An analysis of security incidents on the internet. *PhD thesis, Carnegie Mellon University*.
9. Lough DL. (2001) A taxonomy of computer attacks with applications to wireless networks. *PhD thesis, Virginia Polytechnic Institute and State University.*
10. Simon H, Ray. (2005) A taxonomy of network and computer attacks, Elsevier, *Computers & Security* 24, 31-43.
11. *Common Vulnerabilities and Exposures (CVE) list*, http://cve.mitre.org/, accessible on April, 2012.
12. K. Kendall. (1999) A Database of Computer Attacks for the Evaluation of Intrusion Detection Systems. *Master's thesis, MIT.*
13. R. Lippmann, D. Fried, I. Graf, J. Haines, K. Kendall, D. McClung, D. Weber, S. Webster, D.Wyschogrod, R. Cunningham, and M. Zissman. (2000) Evaluating Intrusion Detection Systems: The 1998 DARPA Off-line Intrusion Detection Evaluation. *Proceedings of the DARPA Information Survivability Conference and Exposition*, pp. 12 – 26.
14. J. W. Haines, L. M. Rossey, R. P. Lippman, and R. K. Cunningham. (2001) Extending the DARPA Off-Line Intrusion Detection Evaluations. *In DARPA Information Survivability Conference and Exposition II*, volume 1, pp. 77 – 88. IEEE.
15. D. Weber. (1998) A taxonomy of computer intrusions. Thesis (S.B. and M.Eng.), Massachusetts Institute of Technology, Dept. of Electrical Engineering and Computer Science.
16. T.F. Gruber. (1993) A Translation Approach to Portable Ontologies, *Knowledge Acquisition*, 5(2): 199-220.
17. S. Staab and A. Maedche. (2000) Ontology Engineering Beyond the Modeling of Concepts and Relations. *Proceedings of the 14th European Congress on Artificial Intelligence*.
18. Deborah L. McGuinness. (2003) Ontologies Come of Age, Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential. MIT Press.
19. J. Undercoffer and J. Pinkston. (2002) An Empirical Analysis of Computer Attacks and Intrusions, *Technical Report TR-CS-03-11*, University of Maryland, Baltimore County.
20. F. Abdoli, M. Kahani. (2008) Using Attacks Ontology in Distributed Intrusion Detection System, *Advances In Computer and Information Sciences and Engineering, 153 - 158.* Springer Science - Business Media B. V.
21. Peng He and George Karabatis. (2012) Using Semantic Networks to Counter Cyber Threats, *Proceedings of IEEE International Conference on Intelligence and Security Informatics, June 11 - 14, 2012 Washington D.C.*
22. The DARPA Intrusion Detection Data Sets by MIT Lincoln Lab, http://www.ll.mit.edu/mission/communications/ist/corpora/ideval/data/index.html, accessible on April, 2012.