Methods and Techniques to Identify Security Incidents Using Domain knowledge and Contextual Information

Ahmed AlEroud
Department of Computer Information Systems Yarmouk University, Jordan
Ahmed.aleroud@yu.edu.jo

George Karabatis
Department of Information Systems, University of Maryland, Baltimore County (UMBC), Baltimore, MD 21250, USA
georgek@umbc.edu

***Abstract*—** a recent trend in intrusion detection is toward utilizing knowledge-based IDSs. Knowledge-based IDSs store knowledge about cyber-attacks and possible vulnerabilities and use this knowledge to guide the process of attack prediction. One significant limitation of knowledge-based IDSs is the lack of contextual information and domain knowledge used to detect attacks. Contextual information is not only the configuration on the targeted systems and their vulnerabilities. It also covers semantic relationships between malicious activities. In addition, domain knowledge extracted from taxonomies about those activities is a significant contextual factor in attack identification. To overcome these limitations, this work introduces a novel contextual framework which consists of several attack prediction models that are utilized in conjunction with IDSs to detect cyber-attacks.

Keywords— Domain knowledge, Data mining, Context, Intrusion detection, Cyber Security.

I. INTRODUCTION

Safeguarding computer systems against attacks is one of the most challenging tasks that cannot be easily measured. Most security mechanisms can be breached due to unknown vulnerabilities and novel hacks applied by attackers to initiate intrusions. The latter has been defined as "any action the user of an information systems takes when he/she is not legally allowed to" [1]. Powell et al. have also defined an intrusion as "a malicious, externally-induced fault resulting from a successful attack" [2]. Halme et al. have referred to an intrusion attempt " as a sequence of actions by means of which an intruder attempts to gain control of a system" [3]. The intrusion detection as a process involves determining that an attack has been attempted to gain unauthorized access to a system. Krugel et al. consider responding to malicious actions that targeted computing and network resources as part of intrusion detection process [4]. Although IDSs have shown a good level of success in detecting intrusion attempts to networks, they still have several research challenges:

First, lack of information about relationships between entities at the prediction time: Integrating the relationships between an entity and other entities in the intrusion detection process is very significant to identify relevant events based on context. In general, events that target the system are not independent. Although the behavior of attackers is not predictable, the sequence of events are initiated to reach to a specific objective, therefore, it is very likely to discover several forms of relationships between those events. Detecting these relationships helps in predicting cyber-attacks at their early stages. Second, existing intrusion detection approaches lack semantic inference and domain knowledge needed to identify cyber-attacks. Recent trends focus on alert correlation and causality analysis to discover these relationships. However, these techniques work at the syntactic not the semantic level. New approaches are needed to convert raw alerts into knowledge with appropriate evidence based on which decisions can be taken. To address these challenges we introduce a new framework that integrates situation and event-based information to create semantic relationships between events that are relevant to that situation. Our framework is driven by several knowledge discovery techniques in addition to domain knowledge extracted from taxonomies to discover attacks. The rest of the paper is organized as the following: Section 2 focuses on our contributions and research motivations. The related work is discussed in section 3. The prediction models in our framework are presented in section 4. Experiments and analysis are discussed in section 5. Finally, the paper is concluded in section 6.

II. CONTRIBUTIONS AND RESEARCH MOTIVATIONS

We introduce a framework with several significant contributions to the cyber-security area. First, we automate the manual and daunting process of human decisions about the possible semantic relationships between security incidents by utilizing contextual patterns in the data. In addition, our approach introduces domain knowledge as a foundation for context-based reasoning in cyber-security, in which defining such relationships is very complicated due to large amount of incoming traffic. The quality of the produced relationships cannot be generated neither by conventional data mining techniques nor by manual investigation performed by domain experts. The produced relationships assist in gaining better understanding of the possible impacts of events that target computer networks. Our methodology significantly enhances these techniques through analyzing situations rather than single events using relational databases as evidenced by our recent research, which has revealed encouraging results attributed to the use of context and domain knowledge [5, 6, 7, 8, 9, 10, 11].

III. RELATED WORK

In this section we discuss the main research approaches and their limitations in reference to context applicability to the current IDSs.

Classification: Classification approaches have been widely utilized in devising signature-based intrusion detection

systems. With reference to contextual information fusion, classification techniques utilize information in *activity* category [12]. The *location* information has been utilized by Sang and Cho in [13] to create a profile-based technique that logs the history of system calls on each host. The host resource usage and the file access events were used to create a decision tree classifier, which is applied at run time to detect suspicious systems calls. The major context modeling techniques in classification approaches are the feature-based profiles which have been applied for attack *prediction* [14] and *filtering* out irrelevant predictions [15].

Clustering: clustering is a learning process that is used to find the structures or patterns in a collection of unlabeled data. A number of clustering techniques have been proposed to detect both known and unknown attacks. The most widely used clustering algorithms for intrusion detection are K-means, DBscan and Self-Organizing Map (SOM). In [16] Meng et al. propose a K-means based intrusion detection technique and test it on an audit network Intrusion detection data set. In [17] a clustering technique called K-map is proposed. The K-map clustering works in the same way as K-means, however, K-map is applied in a multilayer hierarchical approach. Clustering techniques rely on the assumption that similar instances belong to the same cluster and they are part of the same neighborhood, therefore, clustering techniques focus on the *relation* aspect of context. *Activity* information is widely used in clustering approaches. In addition, *Time* information has been utilized to create clustering-based attack detection techniques. For instance, an Eigen space clustering approach which uses time sequence of graphs is proposed and used by Ide et al. [18] to discover cyber-attacks. The technique utilizes the principal eigenvector to partition the graph, then it derives a probability distribution for an anomaly measure that is defined for a time-series data. *Context profiles* which store information about nodes and their clusters is the main context modeling technique used in clustering approaches. Contextual information in clustering approaches has been utilized mainly to support the *prediction* of attacks. Few approaches [19, 20] utilize such information in *filtering* the predictions of IDSs.

Anomaly Detection: The purpose of anomaly detection techniques is to target events that fall outside of the region of predefined sets of benign activities. Chandola et al. [21] define anomalies as "patterns in data that do not conform to a well-defined notion of normal behavior". Several Anomaly detection techniques have been utilized to identify intrusions. For instance, the one class Support Vector Machines (SVM) has been applied in several works as an intrusion detection technique [22, 23, 24]. The attacks are detected by determining which point lies in a sparse region of the feature space. While the major focus of approaches in this category is the *activity* contextual information, there are few of them that focus on other aspects. In [23] Ma et al. utilize *time* aspect to discover anomalies in time series data. An algorithm for anomaly detection from time-series data based on one-class SVM is proposed. The time-series are converted into a set of vectors in the projected spaces. On the same venue, Shon et al. in [25] utilize the temporal relationships between flow of packets during data preprocessing. The temporal relationships among the inputs are used in a SVM learning process that is utilized in detecting unknown attacks. The major usage of contextual information in one class SVM is to *predict* attacks. However, Shon et al. in [25] use context-based packet *filtering* scheme to recognize unknown attacks. They utilize Passive TCP/IP Fingerprinting (PTF) in order to reject incomplete network traffic which violates the TCP/IP standards for policy generation inside well-known platforms.

IV. Tasks and Prediction Models

To address the limitations discussed earlier, we created a framework that works on top of existing rule-based IDSs such as Snort to improve their effectiveness. The major activities performed to create and use our framework are shown in figure 1.

First, using similarity between existing cyber-attacks, we utilize semantic inference to create **contextual relationships** between them. Attacks are represented as nodes on Semantic Link Networks (SLNs). Since a substantial amount of security-based contextual information is organized in ontologies or taxonomies. Extracting such a background knowledge and fusing it in attack prediction models is one of the main tasks in this work. The identified relationships are used at real time to detect contextually related attacks.

Second, we utilize the contextual aspects introduced earlier to profile contextual information as attack profiles that have the capability to filter-out some predictions based on context. The created prediction models are utilized in a layered manner at real time on incoming activities to perform several tasks, such as expanding and filtering-out the predictions of IDSs to improve the detection rate of attacks.

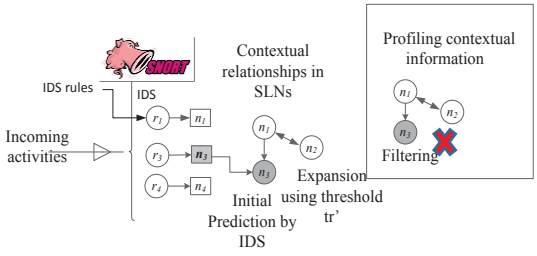

Fig. 1. An overview of attack prediction models in our framework

A. Framework Applicability to TCP connections

One of the main characteristics of an effective IDS is to analyze raw and aggregated data. This part of our work emphasizes creating contextual prediction models to analyze TCP connections that represent traffic generated in a given network. Each connection is a sequence of TCP packets to and from particular IP addresses. TCP packets can be assembled into connection sessions using some software packages such as Bro [26]. TCP suspicious and benign connections contain features that identify activities that target a specific host. The features of TCP connections are divided

into three categories: basic features, content features, and traffic features. The basic features encapsulate all the features that are extracted from TCP/IP packet headers such as, protocol service, flag, wrong_fragment etc. Content features are necessary to detect some attacks by looking for suspicious behavior in the data portions, such as the payload of the original TCP packets, the number of failed logins, whether a root shell is obtained, etc.

Traffic features are those that are computed with respect to a time interval, such as examining only the connections in the past n seconds that have the same destination host as the current connection, and calculating statistics related to protocol behavior, service, etc. Traffic features are also called time-location based traffic features since they are calculated with respect to a specific time-interval and based on the targeted locations.

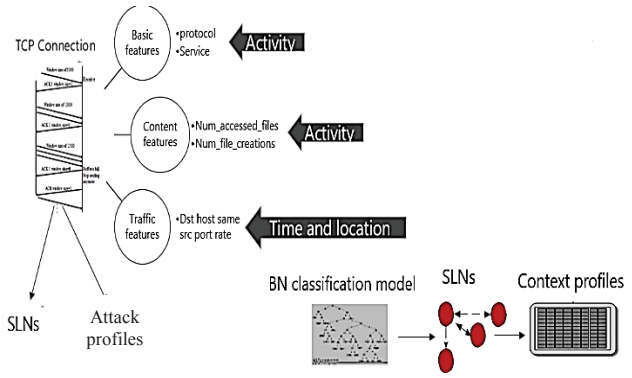

Fig. 2. Identifying Attacks in TCP Connections

Most of basic and content features in the connection sessions describe the activity element of contextual information. Traffic features represent aggregated statistics about the characteristics of connections in regard to time and location contextual aspects. Each of these connections results in consequences, such as specific attacks that lead to damaged, stolen data, or invasion of privacy. On the contrary, the connection can be a legitimate user activity that leads to benign read and write operations on the targeted system.

As shown in figure 2, two prediction models are created in a pre-processing phase and used at run time to predict attacks given unlabeled connections. Activity, time and location features are utilized to create contextual semantic relationships between attacks on Semantic Link Networks (SLNs). Activity features are also used to create the attack profiles (APs) for attacks found in the connection sessions. Details on algorithms and techniques used to create SLNS are described in our previous work [9]. Figure 3 gives an example of an SLN with several attacks. Weights on the edges between nodes represent the similarity values (denoted by *relevance scores* (rs)). Those similarity values are calculated based on feature similarity. SLNs are applied on top of a prediction model *m* created using Bayesian Network classification technique.

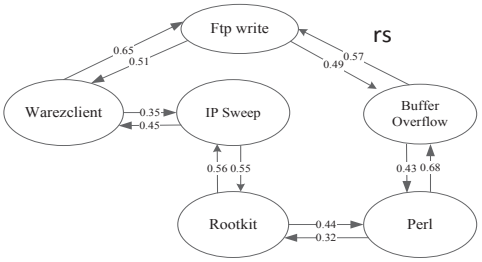

Fig. 3. An SLN example for some security incidents

The BN classifier investigates incoming unlabeled connections and produces an initial prediction, which is either a specific type of an attack or a benign activity. SLNs are then applied as follows. The relationships produced between attacks using SLNs are utilized to expand the initial prediction to include other semantically relevant ones. Since the similarity-based semantic network is confined to the subjective knowledge hidden within the dataset, we adjust it to generate a rule enforced SLN using taxonomies that identify some forms of relationships between nodes. Specifically, we update the relevance scores of the first mode SLN based on relationships between nodes in a taxonomy such as the one shown in figure 4.

B. Domain Knowledge to Improve the quality of contexual Relationships

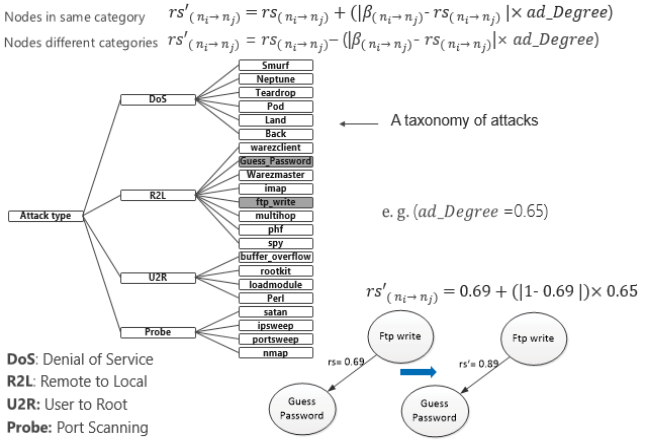

Fig. 4. SLNs with domain Knowledge

We created specific rules to adjust the relevance score $rs_{(n_i \to n_j)}$ between nodes n_i and n_j. The main criterion to adjust relevance scores is: *If any two nodes n_i, n_j have the same parent node (belong to the same category) in a taxonomy, the $rs_{(n_i \to n_j)}$ needs to increase. Otherwise, $rs_{(n_i \to n_j)}$ needs to decrease.* An example on adjusting the relevance score using domain knowledge is given in figure 4. When both nodes n_i and n_j fall in the same category (e.g. Guess password, ftp_write), a taxonomy semantic score ($\beta_{(n_i, n_j)} = 1$). The relevance score $rs_{(n_i \to n_j)}$ is adjusted by adding $|\beta_{(n_i, n_j)} - rs_{(n_i \to n_j)}| \times ad\_Degree$ to its value

resulting in a rule enforced relevance score $rs'_{(n_i \to n_j)}$, where *ad Degree* is the degree of adjustment identified by a domain expert. This adjustment is performed since both nodes are taxonomically similar (i.e. fall in the same attack category). When nodes fall in different categories ($\beta_{(n_i, n_j)} = 0$) i.e., they are taxonomically dissimilar, and $rs_{(n_i \to n_j)}$ is adjusted by subtracting $|\beta_{(n_i, n_j)} - rs_{(n_i \to n_j)}| \times ad\_Degree$ from its value. The procedure we followed to identify the value of *ad_Degree* is described in [28].

C. Creating Attack Profiles

Attack profiles identify the pre-conditions needed to predict attacks based on their features. Attack profiles are applied to the predictions made by SLNs to decrease the side-effect of the expansion process which may lead to some incorrect predictions. Attack profiles are created for each attack node n_i using two set of connection features.

- $G_{N'}$: the features which give the lowest conditional entropy (Global conditional entropy) values when conditioned on a set of related attacks $N' = \{n_1, \ldots, n_k\}$ that belong to the same category of attacks in the corresponding taxonomies.
- L_{n_i} The set of features which give the lowest conditional entropy (local conditional entropy) values when conditioned on a specific attack $n_i | n_i \in N'$. The process of creating attack profiles is summarized in figure 5 and more details are in our previous work [9].

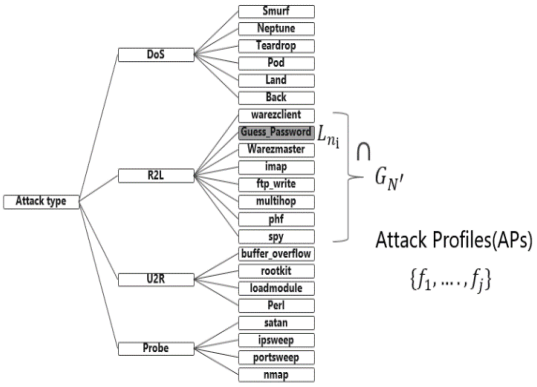

Fig. 5. Generating Attack Profile for password guessing attack

Using this technique if two attacks are highly similar in terms of their features, there is a high likelihood that they are contextually related. Therefore, when their profiles are created, the features utilized to create such profiles should reveal such relationships. This overlap between attack profiles is very important since it preserves the contextual relationship identified between attacks using SLNs. The difference here is that these profiles capture relationships between nodes (e.g., attacks) at the feature-level. They are used to remove the unnecessary relationships produced by SLNs, that is, if two attacks n_i, n_j are known to be non related attacks, and SLNs create a relationship between them, the profiles of attacks n_i, n_j need to be dissimilar to lower the probability of predicting them together.

V. Experiments and Evaluation

In this section we describe the results of several experiments that measure the effectiveness of our prediction models when applied to identify known attacks in TCP connections.

Implementation, dataset, and experiment settings: We developed a prototype system which includes the implementation of our prediction models using an Oracle database. Several data mining tools are used to perform the pre-processing steps in our approaches. In particular, we used Weka [29] and Knime [30] data mining tools to perform preprocessing tasks such as feature selection along with extracting the probabilities of attacks and benign activities to create the BN-based classifier. We utilized PL/SQL to implement several other pre-processing steps such as the creation of feature vectors and SLNs. We utilized DARPA intrusion dataset in our experiments. The number of connections from each category of activities is shown in table I. Two sets of connections are utilized in our experiments, the first one is used for training and the second is for testing.

Tasks on the training data

1) Create a classifier m (a Bayesian Network classifier) that is used to produce an initial prediction using the testing data (incoming connections).
2) Generate $SLNs$ using Anderberg (AD) and Pearson Correlation (PC) similarity measures[28].
3) Generate $SLNs$ using domain knowledge.
4) Generate APs using activity features of connections (e.g. number of packets).

Table I: Connections Used In Experiments From Darpa Intrusion Detection Dataset

Category	Connections to create the prediction models	Connections to test the prediction models
Benign	147,250	110,620
Probe	4,112	4,171
DoS	391,458	300,853
R2L	1130	16,354
U2R	50	69

Tasks on the testing data

1) Classify incoming connections and measure the results in terms of Precision (P), Recall (R), and F measure (F).

$$P = \frac{TP}{TP + FP} \quad (1) \qquad R = \frac{TP}{TP + FN} \quad (2) \qquad F = \frac{(1 + \beta^2) \times PR \times DR}{\beta^2 \times (PR + DR)} \beta^2 = 1 \quad (3)$$

TP, *FP*, and *FN* represent the true positives, false positives, and false negatives respectively. A *TP* occurs when a specific incoming connection is correctly recognized by the prediction model as an attack. TPs for a connection labeled as an attack are expected to be the actual attack (the label) and the attacks that are contextually related to it (e.g. both attacks target ftp application). The latter are identified based on many real world attack scenarios described in the common vulnerabilities exposure (CVE) database [31]. A *FP* occurs when a specific connection under evaluation is incorrectly

recognized as an attack. A *FN* occurs when a specific incoming connection is incorrectly recognized by the system as a benign activity, but in reality it is an actual attack.

2) Expand the initial prediction using AD, PC based SLNs and the $SLNs$ created with domain knowledge to include the top N relevant nodes (e.g., attacks in the same category of the initial prediction).

3) Filter-out non relevant predictions using APs.

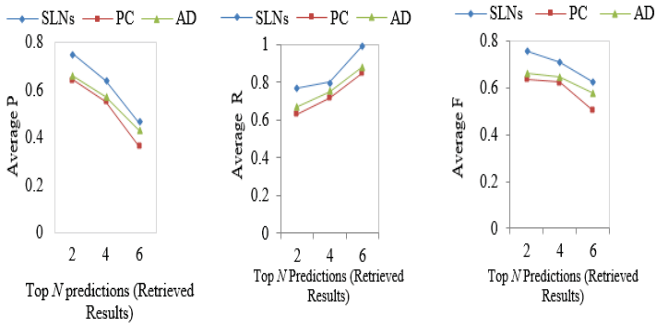

Fig. 6. Average Precision (a), Recall (b), and F (c) for AD, PC, and SLNs when top N predictions are retrieved to expand the initial prediction

The results of this experiment is shown in figure 6. As the number of retrieved predictions increases, the precision decreases due to inclusion of some non-relevant predictions. (figure 6a). Additionally, the figure shows that the precision, Recall, and F-measure values for the Semantic Link Networks created using the domain knowledge (SLNs line) in figures 6a, 6b, and 6c are better than those created using (PC) and (AD) similarity measures without domain knowledge. In the second Experiment, we utilize a new subset S_2 of connections from the DARPA intrusion detection data in addition to the main subset S_1 utilized earlier in our first experiments. The majority of connections in S_2 are benign activities and fewer attack connections that belong to three categories of attacks as shown in table II. This subset has been extracted from the TCP dump format of DARPA dataset by Perona et al. [32] and it consists of attacks in the categories DoS , $R2L$, and $U2R$. There are no flooding (probe) attacks in this subset. The values of precision, recall, and F-measures are reported on figures 7a, 7b and 7c. The figures clearly illustrate better results for $SLNs$ and APs when more attack connections are used to create them. P, R, and F values are higher when the subset S_1 is used ($SLNs + APs\_S1$). We observe relatively lower Precision and Recall results with S_2. The Reason is related to the strength of relationships between benign and attack nodes in the SLNs created using few attack connections (subset S_2). In these networks, attacks and benign activity nodes are not well-separated. While the SLNs created using domain knowledge are expected to lower the probability of this problem by adjusting relationships between nodes, some false positives are still expected resulting in lower precision values. Finally, we compared our results with several conventional classification techniques. To ensure that our comparison is consistent with existing approaches, we used the same evaluation measures (TP and FP rates) utilized in evaluating these approaches. We compared the results of our experiments with several techniques. The results show that using SLNs and APs achieve relatively better results than most approaches as shown in table III.

Table II: Data used for experiments on SLNs and *APs* created using domain knowledge

	Category	Attack type	# of connections
S2	(R2L)	Warezclient,Guess_Password, Warezmaster,Imap, Ftp_Write, Multihop, Phf, Spy	8723
	(DoS)	Smurf, Neptune, Back, Teardrop, Pod, Land	736457
	(U2R)	Buffer_Overflow, Rootkit, Loadmodule, Perl	91
	Probe	Satan, Ipsweep, Portsweep, Nmap	6484
	Benign	-	**544102**
S1			
	(R2L)	dict, dict_simple, ftp_write, guest, multihop, phf, spy, warez,warezclient, warezmaster	2723
	(DoS)	land, syslog, teardrop	1124
	(U2R)	eject, eject-fail, ffb, ffb_clear, format_fail, format_clear, format, imap, load_clear, load_clear, load_module, perl_clear, perlmagic, rootkit	81
	Benign	-	**174873**

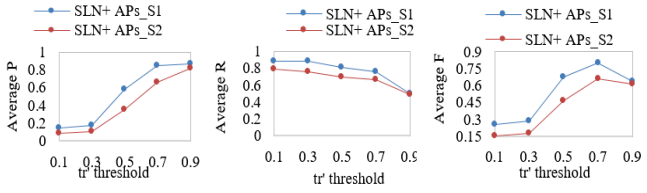

Fig. 7. Average Precision (a), Recall (b), and F (c) for SLNs+APs using two subsets of connections S_1 and S_2

Table III: The effectiveness of SLNs+APs versus other approaches

Approach	Metric	Probe %	DoS %	R2L %	U2R %	Avg %
SLNs & APs	TP	**96.600**	**98.300**	**33.900**	**88.900**	**79.675**
	FP	**0.180**	**0.100**	0.250	0.050	0.305
Multi-Classifier	TP	88.700	97.300	9.600	28.800	56.100
	FP	0.400	0.400	0.100	0.400	0.325
Multi-Layer Perception	TP	88.700	97.200	5.600	13.200	51.175
	FP	0.400	0.300	0.010	0.050	0.190
Gaussian Classifier	TP	90.200	82.400	9.600	22.800	51.250
	FP	11.300	0.900	0.100	0.400	3.175
K-Means Clustering	TP	87.600	97.300	6.400	29.800	55.275
	FP	2.600	0.400	0.100	0.400	0.875
Nearest Cluster Algorithm	TP	88.800	97.100	3.400	2.200	47.875
	FP	0.500	0.300	**0.010**	**0.001**	**0.203**

VI. CONCLUSIONS AND FUTURE WORK

We introduced a context-domain knowledge driven framework that has been implemented and applied in the discovery of cyber-attacks. After a comprehensive research review about contextual information, we found a common classification of the contextual aspects that should be considered in IDSs to make them aware of the current

context. Our approach introduces domain knowledge extracted from taxonomies as a foundation for context-based reasoning in cyber-security. There are several areas where this work can be extended. First, the relationships created using SLNs are between nodes of similar types. Due to multiple dimensions of contextual information, the proposed methodology can be extended to include relationships between nodes of different types, such as creating relationships between attacks and the vulnerabilities that cause them. The identification of relationships among nodes of different types such as attacks and vulnerabilities is very important to identify possible paths of unknown attacks. Second, while our approach can identify the possible relationships between attacks it does not specify the order of their occurrence. As an extension to our framework, the temporal aspect might be considered during the attack prediction process. Third, the current approach requires an initial prediction to be made by a particular intrusion detection technique and then it uses the contextual knowledge produced to extend such a prediction.

VII. ACKNOWLEDGMENT

This line of research has been supported by grants from the State of Maryland-TEDCO (MII), and Northrop Grumman Corporation, USA. The authors would like also to thank the PREDICT team for providing experimental data.

REFERENCES

[1] A. K. Jones and R. S. Sielken, "Computer System Intrusion Detection: A survey" Computer Science Technical Report, Department of Computer Science, University of Virginia, 2000

[2] D. Powell and R. Stroud, "Malicious and Accidental Fault Tolerance for Internet Applications Conceptual Model and Architecture," Technical report series, University of Newcastle Upon Tyne Computing Science, 2001.

[3] L. R. Halme, "AIN'T Misbehaving A taxonomy of Anti-Intrusion Techniques," *Computers and Security,* vol. 14, no. 7, p. 606, 1995.

[4] C. Kruegel, F. Valeur, and G. Vigna, *Intrusion Detection and Correlation: Challenges And Solutions,* vol. 14: Springer, 2004.

[5] A. AlEroud and G. Karabatis, "Discovering Unknown Cyber Attacks using Contextual Misuse and Anomaly Detection," *ASE Science Journal,* vol. 1, no. 1, pp. 106-120, 2013.

[6] A. AlEroud and G. Karabatis, "A System for Cyber Attack Detection Using Contextual Semantics," in *7th International Conference on Knowledge Management in Organizations: Service and Cloud Computing*, Salamanca, Spain, 2012, pp. 431-442.

[7] A. AlEroud and G. Karabatis, "A Contextual Anomaly Detection Approach to Discover Zero-day Attacks," in *ASE International Conference on Cyber Security*, Washington,. USA, 2013, pp. 40 -45.

[8] A. AlEroud and G. Karabatis, "Toward Zero-Day Attack Identification Using Linear Data Transformation Techniques," in *IEEE 7th International Conference on Software Security and Reliability (SERE'13)*, Washington, D.C., 2013, pp. 159-168.

[9] A. Aleroud, G. Karabatis, P. Sharma, and P. He, "Context and Semantics for Detection of Cyber Attacks," *Int. J. Inf. Comput. Secur.,* vol. 6, no. 1, pp. 63-92, 2014.

[10] A. AlEroud and G. Karabatis, "Context Infusion in Semantic Link Networks to Detect Cyber-attacks: A Flow-based Detection Approach," in *Eighth IEEE International Conference on Semantic Computing*, Newport Beach, California, USA 2014.

[11] A. AlEroud and G. Karabatis, "Detecting Zero-day Attacks using Contextual Relations," in *Ninth International Knowledge Management in Organizations Conference*, Santiago, Chile, 2014.

[12] S. Peddabachigari, A. Abraham, and J. Thomas, "Intrusion Detection Systems Using Decision Trees and Support Vector Machines" *International Journal of Applied Science and Computations,* vol. 2, no. 1, pp. 18-134, 2004.

[13] J. H. Sang and S. B. Cho, "Combining Multiple Host-Based Detectors Using Decision Tree," in *Proceedings of 16th Australian Conference on Artificial Intelligence*, Perth, Australia, 2003, pp. 208-220.

[14] Y. Bouzida, F. Cuppens, N. Cuppens-Boulahia, and S. Gombault, "Intrusion Detection Using Principal Component Analysis," in *In Proceedings of the 7th World Multiconference on Systemics, Cybernetics and Informatics*, Orlando, USA, 2004.

[15] X.-B. Li, "A Scalable Decision Tree System and its Application in Pattern Recognition and Intrusion Detection," *Decis. Support Syst.,* vol. 41, no. 1, pp. 112-130, 2005.

[16] J. Meng, H. Shang, and L. Bian, "The Application on Intrusion Detection Based on K-means Cluster Algorithm," in *International Forum on Information Technology and Applications(IFITA '09)*, Chengdu, China., 2009, pp. 150-152.

[17] S. T. Sarasamma, Q. A. Zhu, and J. Huff, "Hierarchical Kohonenen Net for Anomaly Detection in Network Security," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics,* vol. 35, no. 2, pp. 302-312, 2005.

[18] T. Ide and H. Kashima, "Eigenspace-Based Anomaly Detection in Computer Systems," in *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Seattle, WA, USA, 2004, pp. 440-449.

[19] H. Shah, J. Undercoffer, and A. Joshi, "Fuzzy Clustering for Intrusion Detection," in *The 12th IEEE International Conference on Fuzzy Systems (FUZZ '03)*, St Louis, MO, USA, 2003, pp. 1274-1278.

[20] C. Te-Shun, K. K. Yen, N. Pissinou, and K. Makki, "Fuzzy Belief Reasoning for Intrusion Detection Design," in *Third International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIHMSP'07)*, Kaohsiung, Taiwan, 2007, pp. 621-624.

[21] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.,* vol. 41, no. 3, pp. 1-58, 2009.

[22] E. Eskin, A. Arnold, M. Prerau, L. Portnoy, and S. Stolfo, "A Geometric Framework for Unsupervised Anomaly Detection: Detecting Intrusions in Unlabeled Data," in *Proceedings of the Conference on Applications of Data Mining in Computer Security.*, Kluwer Academics, 2002, pp. 78-100.

[23] J. Ma and S. Perkins, "Time-Series Novelty Detection Using One-Class Support Vector Machines," in *Proceedings of the International Joint Conference on Neural Networks.*, Portland, 2003, pp. 1741-1745 vol.3.

[24] L. Kun-Lun, H. Hou-Kuan, T. Sheng-Feng, and X. Wei, "Improving One-Class SVM for Anomaly Detection," in *International Conference on Machine Learning and Cybernetics.*, Xi'an, China, 2003, pp. 3077-3081 Vol.5.

[25] T. Shon and J. Moon, "A hybrid machine learning approach to network anomaly detection," *Information Sciences,* vol. 177, no. 18, pp. 3799-3821, 2007.

[26] V. Paxson. The Bro Network Security Monitor: A Network Security Tool, Lawrence Berkeley National Laboratory. 2013. Version 2.2

[27] S. J. Stolfo, W. Fan, W. Lee, A. Prodromidis, and P. K. Chan, "Cost-based Modeling for Fraud and Intrusion Detection: Results from the JAM Project," in *Proceedings of DARPA Information Survivability Conference and Exposition (DISCEX'00)*, Hilton Head Island SC, 2000, pp. 130-144.

[28] A. AlEroud, "Contextual Information Fusion for the Detection of Cyberattack.," University of Maryland, Baltimore County, http://gradworks.umi.com/36/68/3668684.html, 2014.

[29] E. Frank, T. Smith, and I. Witten. Weka A machine Learning Software, Machine Learning Group at the University of Waikato. 2014. http://www.cs.waikato.ac.nz/ml/weka/.

[30] B. Wiswedel, P. Ohl, and T. Gabriel. KNIME: Kontaz Information Miner. 2014. http://www.knime.org/.

[31] S. Lawler and P. Meunier. (2012, 10/07/2014). *Common Vulnerabilities and Exposures*. Available: http://cve.mitre.org/

[32] I. Perona, I. Gurrutxaga, O. Arbelaitz, J. I. Martín, J. Muguerza, and J. M. Pérez, "Service-Independent Payload Analysis to Improve Intrusion Detection in Network Traffic," in *Proceedings of the 7th Australasian Data Mining Conference*, SA, Australia, 2008, pp. 171-178.