Vol. 5(15) Apr. 2015, PP. 2135-2150

Contents list available at IJMEC

International Journal of Mechatronics, Electrical and Computer Technology (IJMEC)

Journal Homepage: www.aeuso.org

PISSN: 2411-6173 - EISSN: 2305-0543

The Impact Analysis of Modeling Errors for projecting cyber attacks

Kourosh Dadashtabar Ahmadi[1]*, AliJabar Rashidi[2] and Morteza Barari[3]

[1,2,3]*Complex of Information and Communications Technology (ICT), Institute for Research on Information fusion (IRIf),Malek-e-Ashtar University of Technology, Tehran, Iran*

***Corresponding Author's E-mail:** dadashtabar@mut.ac.ir*

Abstract

One of the important components in predicting a credible future for cyber-attacks is the use of the number of attacks. Therefore, any change in the number of attacks will lead to errors in calculating the probabilities. In this paper, the impact of missing alerts on the predictive performance of Variable Length Markov Model for projecting the cyber-attacks is studied. By developing a comprehensive experiment, the impacts 0f missing alerts on the prediction have been obtained by removing the alerts from different locations of the attack sequence in different states. The results of the experiment show that if missed alerts are from just one part of the sequence they will cause less change in prediction accuracy and if missed alerts are scattered throughout the entire sequence, they will cause more change. When the sequence has a smaller symbol space, relatively less change is occurred in prediction accuracy while having larger symbol space causes more change. Overall, the results represent the strengths and weakness of Variable-Length Markov Model in projecting cyber-attacks. Based on this error analysis, a network analyst can infer and assess the predictive performance of Variable Length Markov Model when intrusion detection system loses some of the alerts. This research is an important step in developing a comprehensive report to assist cyber-attacks analysts.

Keywords: *Cyber defense, predictive business, Variable length Markov model, projecting cyber attacks*

1. Introduction

Currently, for effective protection of network centric infrastructures against the increasing number of cyber-attacks and reduction of their impact, in addition to firewalls, intrusion detection systems are used which monitor all activities to detect symbols of suspicious behaviors. An intrusion detection system monitors network traffic to detect suspicious activities and reports the results to the network administrator. "Pattern recognition" and" unusual behavior detection" are two common approaches used in intrusion detection sensors. Intrusion detection systems based on pattern recognition, monitor the packets on the network and compare them with the database which consists of meaningful patterns and in case of a matching pattern discovery, issue an alert. Also, Intrusion detection systems based on detection of unusual behaviors, will announce the observed mismatch by generating a normal profile. One of the problems in using the mentioned systems is generating the flood of alerts that can lead to reduce the network analyst's performance. In order to eliminate and reduce unnecessary alerts, new techniques are used which are called "alert tracking" techniques. By these techniques, a comprehensive report of the alerts will be provided and the alerts which are related to each other are grouping together into the groups called "attack tracks". Using these techniques, together with development of cyber situational awareness, it is possible to develop

Article History:

Received Date: Jan. 03, 2015

Accepted Date: Mar. 26, 2015

Available Online: Apr. 11, 2015

cyber-attacks detection systems with high reliability to detect, track and assess the cyber space situation which is affected by complex and various attacks [22]."Attack identification", "discovery of the relations between attacks(correlating the attacks)" and " predicting the impact of the attack " are effective in situational awareness and prediction of cyber threats in mass volume of data obtained from different sources" as it is shown in Figure 1.

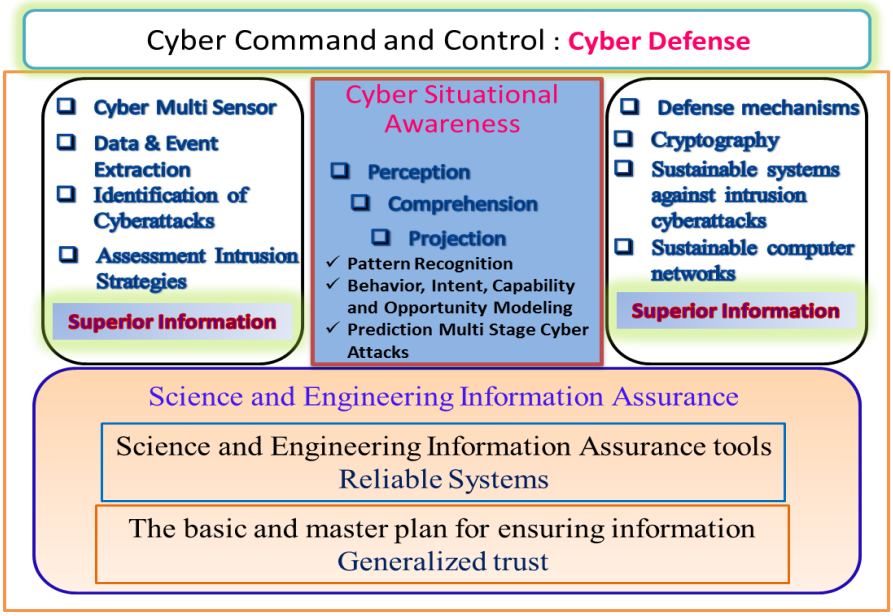

Figure 1: An overview of a cyber-defense system with separation of the domains

Situational awareness is presented in three levels of perception, comprehension and projection. In the first level, perception of symbols is a critical issue and without a basic understanding of important information, the possibility of forming a false picture of a subject highly increases. On this level, the question is what the present realities are. In the second level, Situational awareness is a concept beyond the perception or just consideration of the information, but also it is about integration of various pieces of information to determine the relationship between them and the user's goals. Overall, this level answers to the question what will happen. In third level, which is the highest level of situational awareness, the ability of prospecting the events related to each situation and the user will be addressed. The ability to project current dynamic events into expected future events, allows for timely decision making [13,5].

With description provided in this section, it can be said that providing a perfect cyber defense requires situational awareness which consists of at least seven aspects as follows [14,12]:

- Awareness of current situation: The purpose is the perception of the situation which contains two parts of identification and recognition. The purpose of identification is to determine the type of attack which should probably answer to the questions such as "Where is the source of the attack?", "Who is the attacker?" and "what is the target of the attack?". Identification is just about identification of "an attack" occurred.
- Awareness of the attack impacts: This aspect of cyber situational awareness also called impact assessment. Impact assessment consists of two parts (1) current impact assessment and (2) future impact assessment. In future impact assessment, the question is if the attacker continuously insists on the attack and his motives for the attack has not changed, how will be its effectiveness in future?[23].

- Awareness of the situation that will be achieved: In this aspect, situation tracking is one of the most important components.
- Awareness of the intruders' behaviors: the most important component of this aspect of situational awareness is the analysis of perception and intend of the intruder(s).
- Awareness of why and how the current situation has been caused is another aspect which requires cause and effect analysis.
- Awareness of the quality and reliability of the collected data items which will lead to wisdom and intelligence of the decisions.
- Probable and credible assessment of the current situation: This aspect of situational awareness has been formed from various technologies for projection of the attackers' activities and actions and the paths likely to be attacked. Perception of intent, opportunity and capability of the attacker is also one of the key axes of the aspect.

Considering the aspects mentioned above and the fact that "situational assessment is a machine function" and "situational awareness is a cognitive function", a situational assessment model appropriate for cyber area is used for reasoning (perception), assessment (comprehension) and prediction of future situation (projection) of cyber-attacks which is shown in Figure 2.

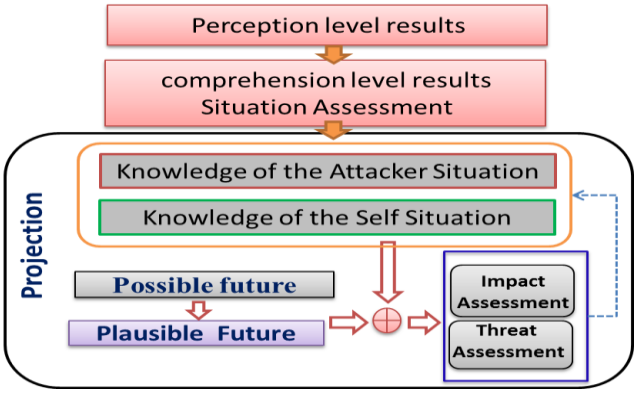

Figure 2: projecting the cyber-attacks in cyber situational awareness

In cyber situational awareness, the attack tracks are converted into the sequences of symbols and consequently the characteristics of the sequences are extracted using sequential modeling schemes. There are two major classes of techniques for sequence characterizations which are Context based Markov Model and State based Hidden Markov Model. Markov chains are the most widely used probabilistic techniques in the area of intrusion detection because they have shown better performance in terms of hit rate and false alarm rate [23]. Daniel Fava [7] implemented Markov models with different orders using a suffix tree and then combined these models into a Variable Length Markov model. Fava used a Variable Length Markov Model and several observed attack tracks to infer the behavior and steps of possible future attacks.

Vidalis [18] has proposed a continually learning and real time system which is able to project the attacks tracks and does not require any prior knowledge about the network structure. In this paper, we propose the use of state based model instead of context based model in the area of projecting cyber-attacks. But in our work, we make use of an algorithm called Causal -State Splitting Reconstruction (CSSR) which had been proposed in [21]. CSSR is preferred over the conventional methods such as Sub-tree merging algorithm presented by Young and Crutchfield or Topologic merging method of Perry and Binder in [1] .The reason for this is that the traditional methods adapt

Markov models with the data but CSSR considers no assumption about the random process and it does not really infer it from its own data. CSSR creates a set of hidden Markov states and statistically is able to produce behavior from its own data. Also, the set of processes that CSSR can present is more than what can be presented by Variable Length Markov. CCSR algorithm is assessed with the dataset which has been already used to test Variable Length Markov Model [23] and eventually the performance of both algorithms will be compared. It should be noted that the mentioned models, compute the probability of future attacks with the assumption that the intrusion detection system detects all the attacks without loss of any alert.

But in fact, even the most efficient intrusion detection systems are not perfect and some of the attacks are likely to stay hidden from their eyes. Since the predictions are made based on the number of previous attacks (symbols), any change in the number of attacks will lead to errors in calculation of the probabilities. A major part of this research is to study the impact of missed alerts on Variable Length Hidden Markov Model predictions in which a "false negative" action has occurred, because the damage caused by a hidden intrusion is much more serious and vital than a "false positive" one. Based on the results of the analysis, the analyst can infer the predictive performance of Variable Length Hidden Markov Model in circumstances that intrusion detection systems miss some of the alerts. This research is effective in the development of a comprehensive report to aid the analyst in assessing the mentioned model. In the present paper, Variable Length Hidden Markov Model [23] was used to evaluate the results.

2. Related works for projecting the attacks

Alerts correlation, alerts aggregation, tracking, projection and impact analysis of cyber-attacks are events which occur after the intrusion detection. The position of each of these events is shown in Figure 3. For large networks, the number of the alerts generated by intrusion detection sensors will be too high. Thus the analyst began to think of using methods that produce a comprehensive report of the alerts. These techniques include alerts aggregation [21], [15], alerts correlation [2], [8], [17] the present and future threat analysis [18], [11]. These techniques are known as the alert tracking in which the related alerts are grouped together. This grouping of data provides network defenders a more complete picture of the traffic on the network under surveillance.

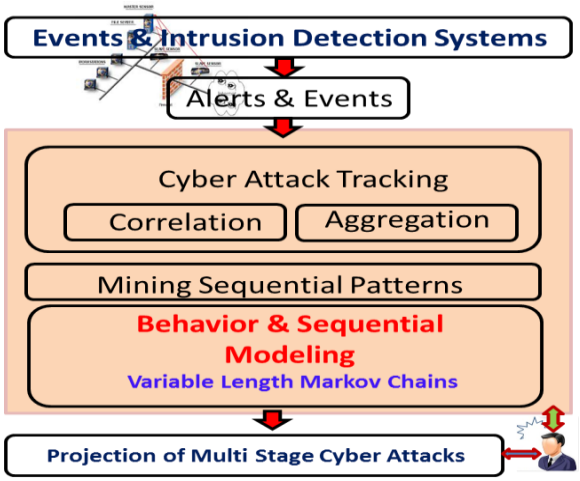

Figure 3: the events after the intrusion detection

Reference [11] has been addressed alert analysis and study, in which in response to an alert generated by the intrusion detection system, a process is implemented to identify the success and failure of the attack. When the attack is not successful, the alert can be eliminated or its priority can

be reduced. This is an effective tool in reducing the number of false alarms that the administrator has to struggle with. The approach proposed in [8] creates the attack scenarios with correlation of alerts in accordance with perquisites and consequences of the attacks. Based on the results of the different types of the attacks, it correlates the alerts with matching the consequences of some previous attacks and perquisites of some future attacks. Reference [17] has introduced an intrusion detection system. The introduced structure consists of three parts including alert aggregation; knowledge based alert evaluation; and alert correlation. The purpose of using this structure is reducing alert overload by aggregation of alerts, reducing false positive alerts, integration of the network and host system information in the form of alert evaluation process, events correlation based on logical relations and producing a global and synthesized report.

The definition of attack correlation has been extended in [4] to correlate the attacks with intrusion goals and to introduce the notion of anti-correlation. This approach provides the analyst a global view of what is happening in the system. This method controls the unobserved actions through hypothesis generation, clusters the iterative actions in a single scenario and also identifies the intruders that often change their intrusion goals and is effective and efficient in detection of the changes of an intrusion scenario. This approach can also be used to remove a bunch of false positive alerts. In [10] a new threat assessment scheme called *TANDI* has been proposed to predict the future attack actions. Tandy predicts future attack actions accurately as far as it is not (part of) a coordinated attack and is not related to inside threats. In [19], even in the presence of abnormal attack events, Tandy was proposed to the network analyst.

This method uses data mining algorithm for any type of attack and some special scenarios of attack to determine their Probabilities of having Following Attacks (PFAS) and then real-time intrusion threats are evaluated using these probability values. In [2] a graph based method has been presented to analyze the network vulnerabilities. This method allows for analysis of both internal and external attacks. The method is able to analyze the risks/threats against a specific network asset and a broad study of possible consequences of a successful attack. This method is based on the idea of an attack graph which contains attack states and possible transitions between them. The attack graph can be used to identify the attack paths that are most likely to succeed and to simulate various attacks. The main advantage of this method over other computer security risk methods is that it considers physical network topology with a series of attacks. Report presented in [11], examined and assessed the applicability of existing security assessment techniques to assess modern cyber threats. In this report, the concepts of risk, threat vulnerability, and threat agent are analyzed and the threat statistics reported from around the world and different sources and the state of the art techniques of threat and risk analysis are reviewed.

2.1. Modelling the projection of cyber -attacks based on Markov Model

Markov model is a probabilistic process on a finite set, $\{s_1,\ldots, S_k\}$ which are usually called the "states". This model, in fact, is a random process in which the distribution of future states depends only on the current state and is independent of the path that has been travelled to reach the current state. According to [20], a random process X(t) is called Markov if for each n, $t_1<t_2<\ldots<t_n$, we have equation(1):

$$P(x((t_n \leq x_n|x(t)\forall t \leq t_{n-1}) = P(x(t_n) \leq x_n|x(t_{n-1})) \quad (1)$$

Zero order Markov Model is equivalent to a polynomial probability distribution. First order Markov Model also has a memory of size "1". A fixed order Markov model means the length of the history or context on which the determination of the values of next state probabilities depends. For example, the next state of the second order Markov Model depends on two previous states. In a Markov process, the current state of the system depends on the previous states. "Finite context" models are also known as Markov models. In this type of model, a probability value is assigned based on the context appears in the symbol. In a "finite context" model of nth order, an event depends on n previous observations. The simplest Markov model is Markov chain in which the states are discrete

and directly observable. Other types of Markov process include hidden Markov process and semi hidden Markov process. On the other hand, "finite state" models known as Hidden Markov Models (HMMS) consist of a visible part (the events) and a hidden part (the states). The events with different probability distributions depend on the system state [19].

A Hidden Markov Process (HMP) is a special type of Markov process in which the generation of observation symbols depends on the emission properties of the states. Therefore, one state usually can generate more than one symbol and we are not directly able to observe the state sequence from the observation sequence. To create a Hidden Markov Model to record the process, there are three important factors for study [9]. The first factor is the determination of the number of states which are really required to cover the process characteristics. The second factor is the choice of the model parameters such as state transition probabilities.

Third factor is the size of the observed sequence. Since if we were limited to a small sequence, we would not be able to estimate optimum parameters of the model reliably. Hidden Markov models for intrusion detection have better performance in terms of hit rate and false alarm rates compared with chi-Square multivariate test and Hoteling's T-square test. The results have been proved using the same testing set of data over all the models in [24]. The hit rate is computed by dividing the total number of hits with the total number of intrusion events the testing data. 100% hit rate and 0% false alarm rate are ideal values which represent the best detection performance by the intrusion detection technique.

2.2. Modeling of projection of the attacks based on Variable Length Markov

Fava in [6] addressed projecting cyber-attacks based on sequential properties of correlated IDS alerts which belonging to multi-stage attack tracks. The proposed methodology consists of three main parts of preprocessing of attack track[1], sequence modelling and prediction. In this work, sequential modelling techniques such as Hidden Markov Model and Variable Length Markov Model were used for interpretation of attack track(s) in the form of a sequence of symbols in the field of projecting the cyber-attacks. Characterization of the sequences of malicious actions was also performed by using finite- context or Markov Models. Markov Models with different orders can be implemented using a Suffix Tree. Training algorithm of the suffix tree was modified by Fava [6] so that a set of finite length sequences is considered instead of a single long sequence of observations. In fava's work [7], the beginning and end of the sequence characters are defined but the real time algorithm proposed by Byer [6] , trains the suffix tree with partial sequences instead of finite completed sequence. Basically, Byer algorithm [6] is more like to a self-learning system which does not requires any preprocessing.

The behavior and steps of the probable future attack can be inferred based on previously observed attacks using Variable Length Markov Model (VLMM) which is formed by blending of different order Markov models. A simple VLMM can attribute fixed weights to any model order. A more complex way is to adapt the weights to give more emphasize and assignment to the higher order models. None of these cases take into account the fact that the relative importance of the models changes depending on their context and its counts. The weights have been derived from escape probabilities. The probability of encountering a previously unseen character is called the Escape Probability. If the probability of an escape at level j is shown with e_j, equivalent weights can be calculated using the relation below:

$$W_j = (1 - e_j) \times \prod_{k=j+1}^{l} e_k \ , -1 \le j \le l \qquad (2)$$

[1] The attacks tracks include regular sets of alerts belonging to a single multi-stage attack

Where L is the highest order context making a non-null prediction. In this formula, the weight of each successively lower order is reduced from one order to the next by the escape probability. The weights are intuitively plausible and correct provided that the escape probabilities are between 0 and 1 and that we would not be able to escape below order -1. Therefore $e_{-1} = 0$. The advantage of expression in terms of escape probability is that they are more easily visualized and understood than the weights themselves, which can become small very quickly. However Fava couldn't show the superiority of any method over others on his research [12].In one of these methods, an additional count is assigned to the number of meeting a context to allow for the occurrence of new characters. This is selected as the default value in Byers [18].

$$e_0 = \frac{1}{C_0 + 1} \tag{3}$$

Using equations of escape probabilities and weights, VLMM estimates the probabilities based on the following relation:

$$P(A) = \sum_{j=-1}^{l} W_j \times P_j(A) \tag{4}$$

Based on the results obtained by Fava [12], the first order model has better performance compared to zero or second or higher order models, because any next event has a strong correlation with the immediate previous event. Moreover, it is shown that because n^{th} order model presents more information which is not obtained by $(n-1)^{th}$ order model, thus VLMM has the best prediction rate.

The above model can also estimate the occurrence of new symbols. When a new symbol occurs, the suffix tree initially is trained as normal and then it is trained again by the suffix history added with the especial new symbol definition.

In [4] a new state based algorithm has been proposed for discovering patterns in data called Causal State Splitting Reconstruction (CSSR). This algorithm creates a minimal set of hidden Markovian states which are statistically able to produce causal states of the process. Causal states are minimal sufficient statistics which are required to predict the future of all lengths. Causal states are not only minimum enough but also unique. These states are prescient, and they form a Markov process by themselves. Each causal state has a unique associated distribution of outputs which is called its "morph". In general every state has a morph, but two states in the same state class may have the same morph. Causal states have another important property that all of their parts have the same morph. Of course each causal state has a unique morph which means none of two causal states have the same conditional distribution of the future. Note that the past and future of a process are independent from each other conditioning the causal states [9].

Current causal states and the next value of the observed process specify the next causal state. All the next symbols have well-defined conditional probabilities. "Transition probability" can be defined as the probability of making a transition from one state to another state while emitting a symbol. The combination Function ϵ from histories to causal states with the labeled transition probabilities is called ϵ machine. Ε machine is deterministic and Markovian in nature meaning that given a causal state at time t, the causal state at time t+1 is independent of the former causal states. Ε-machine reconstruction is any procedure that given a process will produce the process's ϵ- machine. A new ϵ-machine reconstruction algorithm called CSSR that improves the previous algorithm. This algorithm is based on the principle of splitting of new states and is practical when absolutely forced on.

Context based algorithms constructs so called Variable Length Markov Models from sequence data. The contexts are considered as the suffixes of histories and the algorithms work with checking long histories and creating new contexts by splitting existing ones based on a threshold. Variable Length Markov Models, like CSSR, do not depend on domain specific information. Each state in a Variable Length Markov Model is displayed with a single suffix and consists of all and only the

histories ending in that suffix. For many processes in which causal states contain multiple suffixes, multiple contexts are required to show a single causal state; therefore VLMMs are generally more complicated than building of Hidden Markov Models in CSSR. The causal state model is minimal like Variable length Markov model if and only if each causal state contains a single suffix.

3. Error analysis modeling in projecting the attack

In the description of specific errors of each process, you may consider two main types of possible errors: Type 1 error: a wrong decision made when a true null hypothesis is rejected. False positive alerts are included in this category. Type II error: a wrong decision which is made when a false hypothesis is not rejected. False negative alerts come under this category. The definitions of these two types of errors will vary depending on the application. In the field of intrusion detection, they can be redefined: a false positive alert is an alert for a non-malicious action and a false negative alert is about the time an attack is underway and no alert is issued about it. False positive and false negative may occur for each intrusion detection system regardless of how powerful and effective it is.

Both of these errors cause various problems in network security. A false positive alert does not cause an intrusion experience in intrusion detection systems and it is occurred as a result of two reasons: The detection mechanism of intrusion detection systems is faulty or the intrusion detection systems have detected an abnormal condition turns out it is not malignant. However false positives may just cause mockery efforts for the security analysts. On the other hand, a false negative alert is a missed attack that may put the networks or computer systems at risk. It is clear that these are unfavorable and every organization is trying to avoid them. Of course no intrusion detection system is able to detect all the attacks. Therefore, the goal is to achieve good covering against the high priority attacks. Various Other reasons can also be the source of a false negative alert. For example, in order to elude the intrusion detection systems, an attack can incorporate obfuscation techniques. Another possibility is to send the traffic over the processing ability to the intrusion detection systems so that they miss necessary packets for attack detection.

If intrusion detection systems raise an alert about something that has not happened yet, it will just cause the waste of resources and time until the network administrator finds out that it was a false alarm. However, if the intrusion detection system cannot detect an ongoing attack, the attack may be too malicious that damage the whole network and hence risking the security [16]. The mechanism proposed at [3] collected more than two thousands of false negative and false positive cases during sixteen months. This mechanism reported that 85-92% of the cases were related to false negatives and 15.7 % of them were related to false positives. Among all false positives, the reason of 91% of them refers to company's management or intrusion detection systems policies but not due to security problems.

Therefore it is necessary to analyze the impacts of false negatives in detail to improve the network security. If the data reported by the Intrusion Detection System contain some false negatives, it will influence both on the sequence model (the suffix tree of Variable Length Markov Model or ε machine of CSSR) that is made and the estimated probabilities which are calculated based on the model. The present research has studied the impact of false negatives and lost alerts on the performance of the predictive model in use. This helps the analyst in model assessment according to its predictive performance in the circumstances of missed alerts.

In the case of using a model such as Variable Length Markov Model, the amount of the impact that a lost alert will have on the suffix tree or the probability estimations would depend on many factors such as the rate of missing alerts and its estimated probability, the position of the missing alert in the sequence, length of the sequence and the symbol space within the sequence. The future alerts are predicted based on the weights and escape probabilities which depend on the occurrence rate of the symbols in the sequence. Therefore, the missing alerts will cause errors in probability

calculations. For a comprehensive analysis, it is required to study in two ways, one with respect to the position of the missing alerts and the other with respect to the occurrence of the missing alert. We randomly remove the alerts during the analysis to evaluate the models. This random removal in design has been selected with respect to the scenarios in which an intruder makes use of obfuscation techniques to confuse the system with attacks at different times and at different places. We will study the impact in terms of the occurrence of missing alerts by defining common and rare alerts and comparing it with the results based on positional analysis. The prediction accuracy of a model such as Variable Length Markov model in addition to dependence with different factors such as occurrence, position, length of the sequence and the symbol space, it will be affected by missing alerts. Depending on the occurrence rate of missing symbols and their probabilities, missing alerts can result in change of the order of symbol probabilities. Of courses, this does not mean that any change in order of the probabilities always affect the prediction performance. If the estimated probabilities of two symbols are close enough together, they are likely to have a change in their orders due to the impact of missing alerts.

Error analysis of CSSR was done in [4] considering the statistical errors that each of the three procedures of the algorithm can generate. Since the' initialize 'phase just sets up the parameters and data structures, no error is generated in this phase. Two types of errors can be made in 'Homogenize' phase. First, it can group the histories with different distributions for the next symbol. Secondly it may fail in grouping the histories which have the same distributions. Let Si and S_j are suffixes in the same state, with counts n_i and n_j . There is always the vibrational distance t such that the significance test will not separate estimated distributions differing by t or less. If we make n_i sufficiently large, the probability will be close to one and the estimated distribution for i will be within t/2 of the true morph and similarly for j.

Therefore, the estimated morphs for both suffixes are within t of each other and will be merged. If a state includes a finite number of suffixes, with obtaining large enough number of samples of each, we can ensure that all of them are within t/2 of the true morph and are within t of each other and thus will be merged. In this case, the risk of improper splitting can be small. If conditional distribution of each suffix is close enough to the true morph, the test will eventually separate the suffixes belonging to different morphs. Since the determination phase always improves the partition with which it starts, there is no chance of merging the histories that do not belong to each other. In short, if the number of causal states is finite and L_{max} is large enough, the probability that the estimated states are not causal states will become too small, for a large enough N. To analyze the bounds of error probability, CSSR uses Chernoff's inequality which states that:

$$P(d(P_n(S^{\rightarrow 1}|S^{\leftarrow L}=s^{\leftarrow L}),P(S^{\rightarrow 1}|S^{\leftarrow L}=s^{\leftarrow L}))\geq t)\leq \sum_{A\in 2^A} 2e^{-8nt^2}=2^{k+1}e^{-8nt^2} \qquad (5)$$

Where An is the average of first n of X_i($X_1, X_2, \ldots, X_N$ are Bernoly's random variables), with success probability of μ . Convergence of CSSR algorithm based on Kolmogorov-Smirnov test (KS test) and Chernoff's inequality is shown as follows:

$$P(|A_{n-}-\mu|\geq t)\leq 2e^{-2nt^2} \qquad (6)$$

Two kinds of errors are likely to occur in CSSR algorithm. (1) it can group the histories with different distributions for the next symbol. (2) it can fail to group histories with the same distributions. The probability that one suffix or more have the distance t with its true morph:

$$q(t,n) \leq \sum_{i=1}^{s} 2e^{-8n,t^2} = 2^{k+1}\, se^{-8mt^2} \tag{7}$$

Where m is the least of n_i and s is the number of the histories that actually observed or the number of the histories that are required to infer the true states. We can put the upper bound for s as follows with maximum number of possible morphs:

p^*:probability of the most improbable string N: length of the sequence, K=constant=2 , t:difference between suffix and it's true morph

$$S \leq \frac{(k^{L+1}-1)}{(k-1)} \tag{8}$$

$$q(t,n) \leq 2^{k+1} \frac{(k^{L+1}-1)}{(k-1)} e^{-8Np^*t^2} \tag{9}$$

4. Simulation of the proposed design

In this research, the performance of Variable Length Markov model and causal state splitting reconstruction model will be compared by implementation of two sets of simulations. One by using true data set with the assumption that there is no missed alert and another one by removing some alerts from that true data set. The prediction accuracy is the percentage of symbols occurring within a set based on a threshold consisting of symbols with highest probabilities according to the selected model. Thus if an observed symbol was one of the top three predictions with the highest probability, it would be included in the top-3 prediction accuracy part. Prediction accuracy is calculated by dividing the number of correct predictions to the total number of predictions made by the model used. Typically, top 3 predictions are considered relevant and so we have used it in our model analysis. In previous work, test data set of the attacks was divided randomly into two halves. One half of the attacks were used for pre- training of the model and the other half to test the predictive performance of the algorithm.

This system produces the symbols, dynamically trains the models for each alphabet definition in parallel and generates the next phase prediction sets for each attack track based on those changing models. This approach facilitates research and study in the field of adaptive quality of the system with new attack scenarios. For the first simulation set, we implement the correct dataset on both different "Variable Length Markov Model" and "Causal State Splitting Reconstruction" algorithms and compare their performance in terms of prediction accuracy. To enable testing of situation awareness tools which developed for detecting and analyzing the attacks on computer networks, a simulation model was proposed in [35]. This simulation model has been made to generate sample cyber-attacks and intrusion detection sensor alert data. This model allow the user to build a computer network and set up and run a series of cyber-attacks on target machines within that network. Intrusion detection sensors which are deployed within the network, issue appropriate alerts based on the traffic that they observe within the network.

To determine the desired scenario, a user interface is used. When a computer network has been created, it is possible to set up an attack scenario to run on the network. Once the simulation is run and the scenario is implemented, the output files containing intrusion detection system alerts are generated and are applicable to test the situational awareness tools. The outcome of this simulation model is a set of intrusion detection alerts which can be used to evaluate the different cyber security tools. To address the needs for ground truth (the action list which is running during each attack), the study makes use of experimental datasets generated by the simulation model which is implemented by RIT's Industrial and systems Engineering. We have used five sets of data including ground truth

without any noise, to evaluate VLMM and CSSR algorithms. The data sets used, the length of the sequence and the number of the symbols in the sequence are displayed in Table1.

Table 1: The dataset with the length of the sequence and the number of symbols in the sequence

Dataset	Symbols	Alarms
BVT_2A	39	311
BVT_2B	35	271
BVT_2F	33	313
BVT_2H	30	228
BVT_2I	38	431

Table 2 shows the prediction accuracies for the data sets without any repetitions. Here it is assumed that the data sets are true data sets, which do not have a missed alert. The accuracies resulted from both VLMM and CSSR algorithms are hopeful and considerable given a relatively large number of symbols in the corresponding symbol space. Notice to BVT-2A where a 3 out of 39 blind guess gives a prediction rate of 7/6% which is much smaller than 36.10% by VLMM and 46.71% by CSSR. If we consider the average of all data sets as it is shown in Figure 4, we can observe that in terms of top1, top2 and top3 accuracies, both algorithms can give similar results. Of course it can be seen from Table (2) that for some cases, there is some significance difference in top1 accuracies of the two algorithms. For example, for data set BVT-2B, top1 accuracy by VLMM is equal to 62.18% whereas the figure for Causal-State Splitting Reconstruction Model is only 6.5%. This is probably because the symbol to which CSSR often assigned the highest probability has occurred less frequently while the symbol to which VLMM assigned the highest probability has occurred more frequently.

Table 2: The prediction accuracy for the dataset without any repetition

BVT-2A	CSSR	VLMM
TOP-3	46.71	36.1
TOP-2	37.01	28.26
TOP-1	23.96	17.89
BVT-2B		
TOP-3	46.97	37.70
TOP-2	35.74	36.99
TOP-1	6.6	18.43
BVT-2F		
TOP-3	41.2	42.28
TOP-2	30.3	34.73
TOP-1	20.59	19.26
BVT-2H		
TOP-3	31.03	32.57
TOP-2	22.23	22.27
TOP-1	10.27	12.83
BVT-2I		
TOP-3	29.08	31.58
TOP-2	19.82	25.70
TOP-1	12.4	14.03

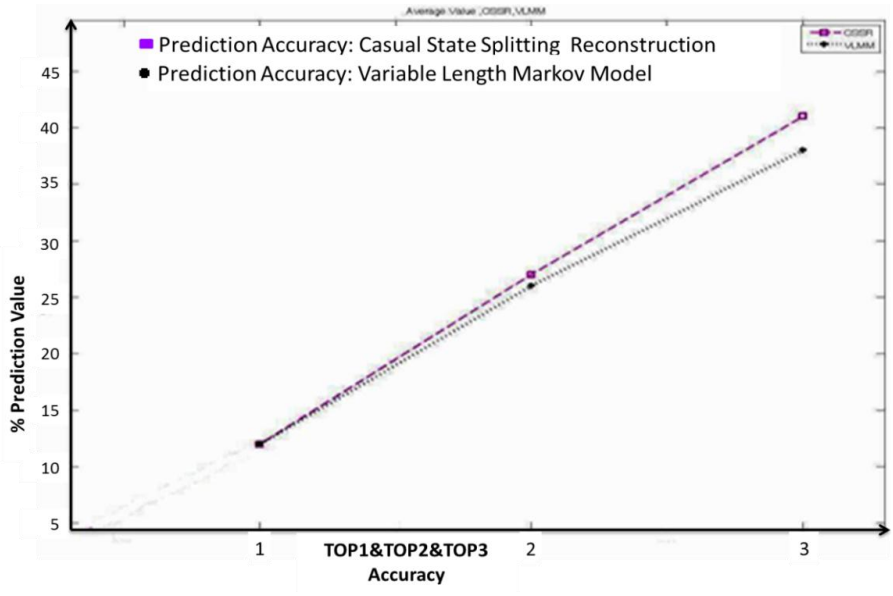

Figure 4: Average prediction accuracies of two models

For the nature of different data sets, the results obtained with two algorithms are not always consistent. Based on the obtained results, we can see that CSSR does not give any better results in terms of predictive performance. Since we need infinitely long sequences to see better performance of CSSR algorithm, whereas VLMM is a more simple process, it is suitable for error analysis of missed alerts. For second simulation set, we analyze the impact of missed alerts using VLMM and by observing the change in prediction accuracies. One way to study the impact of missing alerts using a particular algorithm is by adding the alerts to the data where we assume that the data gathered from the simulation model is a result of inability of intrusion detection system in detection of the alerts. Of course this method is vague and complicated because many factors such as different characteristics of the alert or introducing a new alert or already occurred alerts must be considered. An easier method is to remove alerts from the data sequence where we assume the data resulted from simulation model is a result of a perfect intrusion detection system with complete detection of all the alerts without missing any attack.

In order to have a more comprehensive analysis, we study the impact of missing alerts based on the position and occurrence. In the process, we repeat the analysis for the sequences of different lengths and different symbol spaces since they would also influence the prediction accuracies. Because the position of the alert is one of the factors which can affect prediction accuracy, first we study the impact of missing alerts in terms of its position. To this end, we decided to divide the whole sequence into three parts and call them as beginning, middle and end parts. Then positions are randomly selected within these three parts and are removed. To receive a complete analysis, we randomly remove some alerts from the entire sequence. Although each data set includes several attack tracks, when removing the alerts, we consider the entire data set as a single sequence which has produced after merging several tracks into one order by VLMM.

The random removal is planned in order to be a reflection of the behavior of some real world scenarios where the intruders attempt to confuse the intrusion detection system using obfuscation techniques and confound the system with different attacks sometimes in the beginning, sometimes in the middle and sometimes in final part of the sequence. We have done the simulation with changing missed alerts percentage to 5%, 25% and 40%. In this study, the beginning part of each sequence is defined within the range from 0 to 40%, the middle part within the range from 30% to 70% and the final part within the range from 60% to 100%. Overlapping is used to allow us to remove any higher percentage of alerts and these percentages are calculated in terms of total length of the sequence. We know that because of missing alerts existence, prediction accuracy of each symbol increases or decreases or remain unchanged. We then calculate the change in the predictions

accuracies for making observations. The used dataset and the accuracy of correct predictions are shown in Table 3.

Table 3: VLMM error analysis dataset

Dataset	Alarms	Symbols	TOP1%	TOP3%
IP1 Destination	311	39	53.5	82.12
IP2 Destination	413	6	51.41	86.33
IP3 Destination	534	6	66.74	89.88

Another important factor is the occurrence rate of missed alerts which affects variable-length Markov model and the prediction accuracies. To analyze based on the occurrence rate, we first identify m first cases (top-m) as common alerts and n last cases (bottom-n) as rare alerts. Selection of the numbers n and m depends on the symbol space in the sequence and varies from one sequence to another. Generally for the sequences with higher symbol space, if L is the length of the sequence, we can consider 5 first cases (top-5) as common alerts and remaining (L-5) as rare alerts. Since we selected top-3 as the threshold for prediction accuracy values, if we define top-5 alerts as common alerts, it will be sufficient to us because a change in symbol orders will also change the prediction accuracy. Therefore if the 5th and 4th ranked symbols are sufficiently close to 3th ranked symbol, the order of these symbols may change due to missing symbols and cause a change in their ranks which in turn will change the prediction accuracy.

This is less likely that a symbol above 5th rank change the order with 3th or less ranked symbol. Thus 5 is a reasonable number to choose for the sequences with larger symbol space. Of course for the sequences with relatively fewer symbols, the numbers will change and we can possibly select top-3 as common alerts and the remaining as rare alerts. This is because for the sequences with fewer symbols, the sequence usually fills with top-3 ranked symbols and hence higher prediction accuracy for these cases. After verifying common and rare alerts, we randomly remove the alerts from the whole sequence for 10 trials. After seeing the new prediction accuracy, we have used the same metric i.e. Change in prediction accuracy which is the prediction accuracy difference with and without missed alerts for different test cases. We have compared these results with the scenarios where random alerts were removed from the whole sequence.

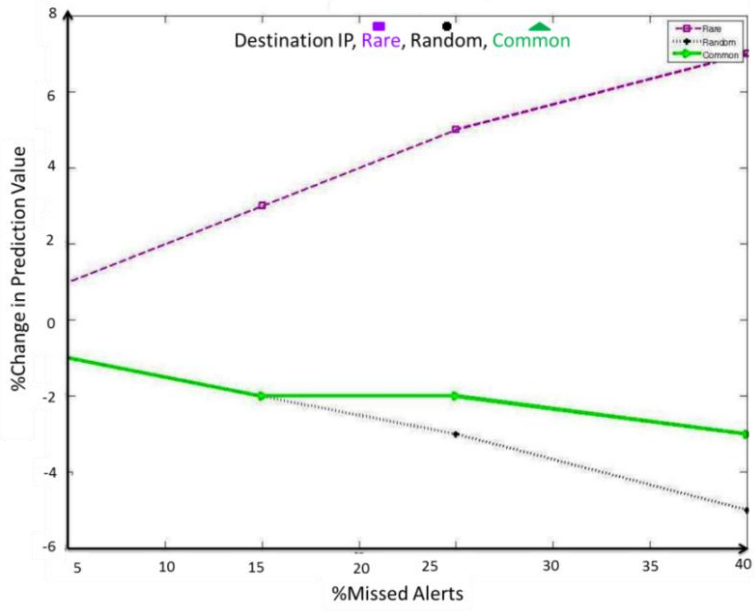

Figure 5: The change in prediction accuracy based on destination IP dataset

Figure 5 shows the change in prediction accuracy for 10 trials in each case for different percentages of missing common and rare alerts. The Figure also shows the results for the scenarios where the alerts are randomly missing throughout the sequence. For all cases of common alerts missing, the prediction accuracy decreased. For all cases of rare alerts missing, the prediction accuracy increased. Since the prediction accuracies top3, top2 and top-1 depend on occurrence rate of the alerts, in most cases, if alert missing actually occurs more frequently in the data, it will reduce the prediction accuracies. If we make our way to a mathematical point of view, this result makes sense. Because if a rare alert is missing when calculating for example top-1 prediction accuracy, the numerator remains unchanged but the denominator is added by one which increases the result. Whereas if a common alert is missing, both numerator and denominator will decrease by 1.

In other words, given that the top-3 true prediction accuracy is equal to X/Y then in case of missing of *n* common alerts(assuming that they all fall under top-3), the new top-3 accuracy will be (x-n)/(y-n) while in case of missing of *n* rare alerts, the new prediction accuracy will be x/(y-n) and clearly x/(y-n)> (x-n)/(y-n). Therefore according to prediction performance of VLMM, it is better for IDS to miss a rare event instead of a more frequent event. When rare alerts are missing, for data sets with attribute "Destination IP", the increase in prediction accuracy is within the range of 1% to 6%. When common alerts are missing, for data sets with attribute" Destination IP", the decrease in prediction accuracy is in the range of 1% to 2%. It can be also seen from Figure 5 that the change in accuracy when common alerts are missing is similar to when the alerts are missing randomly. Because the sequences mainly include common alerts, when we remove the alerts randomly, it is more likely that randomly removed alerts fall under common alerts.

It can be also seen that the magnitudes of change for rare alerts missing case are higher than random and common alerts missing case. The reason is that it is more likely for random alerts to have a combination of common and rare alerts, and since one tries to reduce the accuracy and the other one tries to increase it, the ultimate change in most cases cannot be too large. For all cases of the rare alerts missing, we can see that the change in prediction accuracy is proportional to the percentage of missing alerts. Since *x/(y-n)* increases with *n*, the change also increases with n. If we try to remove any further alerts beyond this point, we will be removing the common alerts not the rare alerts. Generally, in rare alerts missing scenarios, the change in prediction accuracy is less for the sequences with less symbol space and more for the sequences with more symbol space. For the sequences with symbol space of around 9-12 symbols, increase in prediction accuracy is in the range of 1%-11% and for the sequences with a symbol space of around 63-77 symbols, increase in prediction accuracy is in the range of 2%-19%.

For the real data (without any alert missing), VLMM and CSSR both provides comparable results however VLMM has a more simple process. For position based analysis, for small percentage of missed alerts (about 5%-10%), the decrease in prediction accuracy is very small and in the range of 0.5%-1%, irrespective of whether the alerts are missing from the beginning, middle or end or in the whole of sequence. When the percentage of missing alerts increase to more than 10%, prediction accuracy gets worse when the alerts are missing in the whole sequence (reduction in the range of 4-7%) compared to when they are at the beginning, middle or final parts of the sequence(reduction in the range 0-3%).

This is probably due to that many sequences will have the alerts with higher weights scattered throughout the whole sequence rather than just in one single part (of three parts of beginning, middle and final). When the common alerts are missing, prediction accuracy reduces and when rare alerts are missing, prediction accuracy increases. When rare alerts are missing, for the dataset with the attribute "Destination IP", the mentioned increase is in the range of 1-6% and when common alerts are missing, for the data set with attribute "Destination IP", the accuracy decreases in the range of 1-2%. The magnitude of change in prediction accuracy is more when rare alerts are missing compared to when common or random alerts are missing. The change in prediction accuracy is less for the sequences with small symbol space and is more for the sequences with large symbol space. For the sequences with symbol space of around 9-12 symbols, prediction accuracy increases in the

range of 1-11% and for the sequences with symbol space of around 63-77 symbols, the increase is in the range of 2-19%. In this study, error analysis was performed using Variable Length Markov Model.

This can be expanded by doing a similar analysis by using Causal-State Splitting Reconstruction Algorithm. As previously mentioned, the error probability of CSSR can be defined as follows: The equation for large values of *N* (the length of the sequence) tends to zero. In other words, with taking a big enough N, the probability that the correct states are inferred will be close to 1. According to [4], if the number of causal states is finite and L_{max} is large enough, the probability that the states estimated are not causal states becomes small for the large enough values of N. Since the performance of the mentioned model is dependent on many parameters, a similar analysis on the model will give different results as compared to VLMM based on the settings.

Because ϵ– machine and suffix tree are two different models, the impact of missed alerts on them will be different and consequently the changes in prediction accuracies will vary. For example if L_{max} is too larger than N, the probabilities of long strings will not find a consistent estimation, so the model tends to produce too many states which might lead to erroneous prediction accuracies. Since the state based model strongly depends on the length of the sequence, the change in its prediction accuracies might be worse for higher percentages of missed alerts compared to VLMM. The percentage of missed alerts and the selected parameters may cause that the state based model not having enough data in which CSSR will not reconstruct a true model.

Conclusion

The main contribution of this research is the use of Causal-State Splitting Reconstruction Model in the field of projecting cyber-attacks and its comparison with Variable Length Markov Model. Because other state based models start with generation of the most complex possible null model and then refine it by merging whereas CSSR works with a zero complexity null model by putting each history in one state and then adds states only if the current set of states are not enough. Moreover the causal states which CSSR generates have important predictive factors and hence CSSR is preferred over other state based methods such as state- merging algorithms. Although CSSR like VLMM does not consider any priori assumption about the structure of the system, it cannot use such information even in case of its existence. The other important part of this research is to study about the impact of missed alerts on Variable Length Markov Model in projecting cyber-attacks. Since VLMM is simpler in terms of building the model and it does not need to have infinitely large sequences, here we used VLMM for the error analysis. With designing a comprehensive experiment, we studied and evaluated the impacts of missing alerts by removing the alerts from different positions of the attack sequence with different rates. The results showed that the prediction accuracy will be low when the missed alerts belonging to one part of the sequence and it will be higher if the missed alerts are throughout the entire sequence. The prediction accuracy increases when rare alerts are missing and decreases when common alerts are missing. Moreover, the change in prediction accuracy is less when the sequence has a smaller symbol space and is more when having a bigger symbol space. This research focused on error analysis caused by missed alerts using Variable Length Markov Model. However a similar analysis can be done by any other model. Selection of a dataset is always an important factor in evaluation of any situational awareness tool. This can be done in two ways: experiment with the data from a simulated network and using data collected from a real network. We took advantage of the simulated network. Because of having limitation in implementation of CSSR, for its comparison with VLMM, we ran the CSSR on a data set with just a little symbol space. It would be interesting to observe the performance of the model when more symbols are available.

References

[1] B. John Argauer, Jr. "Virtual Terrain Assisted Impact Assessment for Cyber Attacks", Rochester, New York. July 2007.

[2] C. Phillips and L. P. Swiler, "A graph-based system for network-vulnerability analysis", in Proceed-ings of the 1998 workshop for new security paradigms, pp. 71 – 79, (New York, NY, USA), 1998.

[3] C. Cipriano, A. Zand, A. Houmansadr, C. Kruegel, and G. Vigna, "Nexat: A history-based approach to predict attacker actions," presented at the Proceedings of the 27th Annual Computer Security Applications Conference, pp. 383–392, 2011.

[4] C. Rohilla Shalizi and Kristina Lisa Klinkner. Blind construction of optimal nonlinear recursive predictors for discrete sequences. In Proceedings of the 20th conference on Uncertainty in artificial intelligence, UAI '04, pages 504–511, Arlington,Virginia, United States, 2004. AUAI Press.

[5] D. Man, Y. Wang, Y. Wu, and W. Wang, "A combined prediction method for network security situation," International Conference on presented at the Computational Intelligence and Software Engineering (CiSE), 2010, pp. 1–4, 2010.

[6] D. S. Fava, S. R. Byers, and S. J. Yang, "Projecting cyber attacks through variable-length markov models," Inf. Forensics Secur. IEEE Trans. On, vol. 3, no. 3, pp. 359–369, 2008.

[7] D. Fava, J. Holsopple, S. J. Yang, and B. Argauer, "Terrain and behavior modeling for projecting multistage cyber-attacks," 10th International Conference in Information Fusion, pp. 1–7, 2007.

[8] F. Valeur, G. Vigna, C. Kruegel, R.A. Kemmerer, "Comprehensive approach to intrusion detection alert correlation", IEEE Transactions on Dependable and Secure Computing 1 (3) 146–169(2004).

[9] F. Gao, J. Sun, and Z. Wei, "The prediction role of hidden markov model in intrusion detection," presented at the Electrical and Computer Engineering, IEEE CCECE 2003, vol. 2, pp. 893–896, 2003.

[10] J. Holsopple, S. J. Yang, and M. Sudit, "TANDI: threat assessment of network data and information," presented at the Defense and Security Symposium, p. 62420O–62420O, 2006.

[11] J. Holsopple, J. Yang, and M. Sudit "TANDI: Threat assessment of network data and information," in Proceedings of SPIE, Defense and Security Symposium, vol. 6242, pp. 114–129, April 2006.

[12] J. Holsopple and S. Yang, "FuSIA: Future situation and impact awareness ",in Proceedings of 11th International Conference on Information Fusion, pp. 1–8, 2008.

[13] J. Wu, L. Yin, and Y. Guo, "Cyber-attacks prediction model based on Bayesian network," presented at the Proceedings of the 2012 IEEE 18th International Conference on Parallel and Distributed Systems, pp. 730–731, 2012.

[14] K. Tang, M. Zhao, and M. Zhou, "Cyber Insider Threats Situation Awareness Using Game Theory and Information Fusion-based User Behavior Predicting Algorithm," *J. Inf. Comput.Sci.*, vol. 8, no. 3, pp. 529–545, 2011.

[15] P. G. Neumann and D. B. Parker, "A summary of computer misuse techniques", in Proceedings of the 12th National Computer Security Conference, pp. 396–407, (Baltimore, Maryland, USA), 1989.

[16] P. Liu, W. Zang, and M. Yu, "Incentive-based modeling and inference of attacker intent, objectives, and strategies," *ACM Trans. Inf. Syst. Secur.TISSEC*, vol. 8, no. 1, pp. 78–118, 2005.

[17] P. Porras, M. Fong, and A. Valdes, "A mission-impact-based approach to infosec alarm correlation" in Recent Advances in Intrusion Detection. 5th International Symposium, RAID 2002. Proceedings (Lecture Notes in Computer Science Vo.2516), pp. 95 – 114, (Zurich, Switzerland), 2002.

[18] S.Vidalis and A. Jones, "Using vulnerability trees for decision making in threat assessment", tech. rep. University of Glamorgan, School of Computing, Wales, UK, June 2003.

[19] S.-H. Chien and C.-S. Ho, "A Novel Threat Prediction Framework for Network Security," in *Advances in Information Technology and Industry Applications*, Springer, pp. 1–9, 2012.

[20] S. J. Yang, A. Stotz, J. Holsopple, M. Sudit, and M. Kuhl, "High level information fusion for tracking and projection of multistage cyber-attacks," Inf. Fusion, vol. 10, no. 1, pp. 107–121, 2009.

[21] U. Lindqvist and E. Jonsson, "How to systematically classify computer security intrusions", in Pro-ceedings of the IEEE Symposium on Security and Privacy, pp. 154–163, (Oakland, CA, USA), 1997.

[22] X. Qin and W. Lee, "Discovering novel attack strategies from INFOSEC alerts," in *Data Warehousing and Data Mining Techniques for Cyber Security*, Springer, pp. 109–157, 2007.

[23] X. Qin and W. Lee, "Attack plan recognition and prediction using causal networks," presented at the Computer Security Applications Conference. 20th Annual, pp. 370–379, 2004.

[24] Z. Li, J. Lei, L. Wang, and D. Li, "A data mining approach to generating network attack graph for intrusion prediction," presented at the Fuzzy Systems and Knowledge Discovery, Fourth International Conference on FSKD 2007, vol. 4, pp. 307–311, 2007.