Knowledge Based Context Awareness Network Security For Wireless Networks

Deepa U. Mishra

Department of Computer Engineering
Smt. Kashibai Navale College of Engineering
Pune, India

***Abstract* - Context awareness network security is an effective solution to the problems, network systems are suffering from, such as worms, virus, network eavesdropping, sniffing etc. Traditional security controls exist but they operate exclusive to one another and cannot provide evaluation of threats. The false positive and negative rates of these devices are too high. Hence it is very difficult to obtain the security state of the whole network. This paper proposes Knowledge based Context Awareness Network Security (KCANS). The security alert events are gathered from various network security situation sensors which are then analyzed and fused using D-S evidence theory as fusion mechanism. Network security context is generated by extracting frequent attack patterns based on knowledge discovery method. Performance analysis shows that the proposed system have improved results in terms of average end to end delay, average energy consumption, packet delivery ratio and throughput as compared to without knowledge discovery technique.**

***Keywords* - network security, situation awareness, knowledge discovery, security event**

I. INTRODUCTION

Intrusion Detection Systems (IDS) have become very popular as mechanism for protecting network systems from various types of malicious activities. However, current IDS generate large volume of alarms every day, maximum of which are false alarms that lead to blocking communication through incoming port connections[1-2]. As these devices have pre-configured rules and according to these rules they react to network behavior. A mechanism to provide Context Awareness Network

Security (CANS) is needed to overcome the problems of the existing systems. Endsley defined Situation Awareness (SA) as “the perception of the elements in the environment within a volume of time and space; the comprehension of their meaning and the projection of their status in the near future”[3]. Four levels of SA have been proposed that includes perception, comprehension, projection and resolution [4]. But SA in computer network security is in advance stages. Bass was the first who introduced this concept into network security and proposed the network security perception frame based on multi-sensor data fusion [5-6].

The remainder of this paper is organized as follows: Section II presents related work, Section III introduces the proposed KCANS model and its sub modules, Section IV provides simulation and evaluation of results and finally Section V concludes the paper.

II. RELATED WORK

To reduce the false positive and false negative rates of IDS one major prerequisite is fusing data from multiple heterogeneous sources in an effective way. Data fusion methods can fulfill the requirement of CANS and many fusion methods have been proposed. Liu et al. [7] proposed a multi sensor data fusion method using multi-class support vector machines as fusion engine. Further heterogeneous alert data fusion can be achieved by using multi-layer feed-forward neural network [8] for generating network security situation.

Evolutionary neural network [9] optimizes the parameters of neural network and extracts the network security situational factors that provide the quantification of network security situation. Other fusion methods such as D-S evidence theory can provide better performance that has to be considered. Juan et al. [10] proposed techniques for alert analysis and evaluation of threats that provide high level knowledge, such as different severity levels for different types of attacks to locate the most dangerous attack. This reduces the administrator's time and potential in processing enormous amount of alert events. It generates attack graphs based on time and space dimension but no mechanism for alert fusion was considered.

The false alarm rate of IDS can be improved by using honeypot [11] but it lacks in providing the security state of the whole network. Zhao-Yang et al. [12] proposed a network security situation evaluation method based on D-S evidence theory as the information fusion technique to improve performance. Fang Lan et al. [13] presented a framework for network security situation awareness based on knowledge discovery method that supports for exact designing and automatic construction of network security situation graph

III. PROPOSED KCANS MODEL AND ITS MODULES

The KCANS model is shown in Fig. 1. The alert events are generated by network security context sensors either because of network attacks or whenever the monitored parameter exceeds the threshold limit. Alert data may be of different formats as they are collected from heterogeneous sensors. Maintaining uniformity is a major issue when dealing with alert event data from different sources.

Each event is initially processed and converted to an identical format. It is represented as a multi-tuple:

e_i={detect Time $_i$,eventType $_i$, attack $_i$,srcIP$_i$, desIP$_i$, srcPort$_i$, desPort$_i$, protocol $_i$, sensorID$_i$}.

A. Data processing Module

The data processing module is designed to perform different operations on the alert event data collected by the agents in order to reduce the number of effective alert events. It includes following operations:

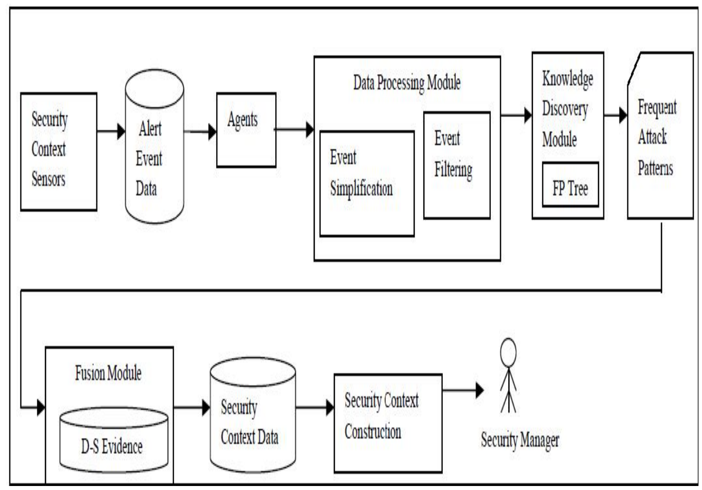

Fig. 1. The KCANS model

Event simplification $[e_1, e_2, \bullet\bullet, e_n] \rightarrow e_m$ the repeated alert events are simplified, which have occurred number of times, to reduce the total number of effective events.

Event filtering $[e_i, P(e_i) \notin H] \rightarrow \varnothing$ if the property P (e_i) does not belong to a specific valid set of H the event is filtered out. Such events occur due to some network conditions and are not associated with any type of intrusion.

B. Knowledge discovery Module

Knowledge Discovery is a method of finding new attack patterns from the set of security alert events gathered from sensors, which are meaningful for security context generation. The frequent pattern (FP) mining algorithm [14 -15] is chosen to extract the security context knowledge from the set of security alert events. The FP tree algorithm is used to generate the frequent patterns of attack. It abbreviates ample databases to compressed tree structure and efficiently mines the set of frequent patterns. The FP tree algorithm executes faster as it avoids repeated database scanning and does not use candidate item generation. It includes tree generation algorithm and FP_growth frequent pattern mining algorithm.

FP_growth adopts a divide and conquer strategy as follows. First, it compresses the database representing frequent items into a frequent –pattern tree, or FP tree, which retains the item- set association information. It then divides the compressed database into a set of conditional databases; each associated with one frequent item or "pattern fragment", and mines each such database separately.

TABLE I
SET OF EVENTS

TID	Event ID
1	E1,E2,E5
2	E2,E4
3	E2,E3
4	E1,E2,E4
5	E1,E3
6	E2,E3
7	E1,E3
8	E1,E2,E3,E5
9	E1,E2,E3

Table I shows set of events occurring during different time intervals. The database is mined as follows. The first scan of the database derives the set of frequent items (1-itemsets) and their support counts (frequencies). Let the minimum support count be 2. The set of frequent items is sorted in the order of descending support count. This resulting set or list is denoted L. Thus L= {{E2:7}, {E1:6}, {E3:6}, {E4:2}, {E5:2}}

An FP tree is then constructed as follows. First create the root of the tree, labelled with "null". Scan the database second time.

The events in each time interval are processed in L order and a branch is created for each time interval. To facilitate tree traversal, an event header table is built so that each item points to its occurrences in the tree via a chain of node links. The tree obtained after scanning all of the time intervals is shown in Figure 2 with the associated node-links.

The FP Tree is mined as follows. Start from each frequent length-1 pattern, construct its conditional pattern base, then construct its conditional FP tree and perform mining recursively on such a tree. The pattern growth is achieved by the concatenation of the suffix pattern with the frequent patterns generated from a conditional FP tree

Mining of the FP tree is summarized in Table II and detailed as follows. First E5 is considered. E5 occurs in two branches of FP tree of figure. The paths formed by these branches are {E2, E1, E5:1} and {E2, E1, E3, E5:1}. Therefore considering E5 as a suffix, its corresponding two prefix paths are {E2, E1:1} and {E2, E1, E3:1}, which forms its conditional pattern base. Its conditional FP tree contains only a single path, {E2:2, E1:2}, E3 is not included because its support count of 1 is less than the minimum support count.

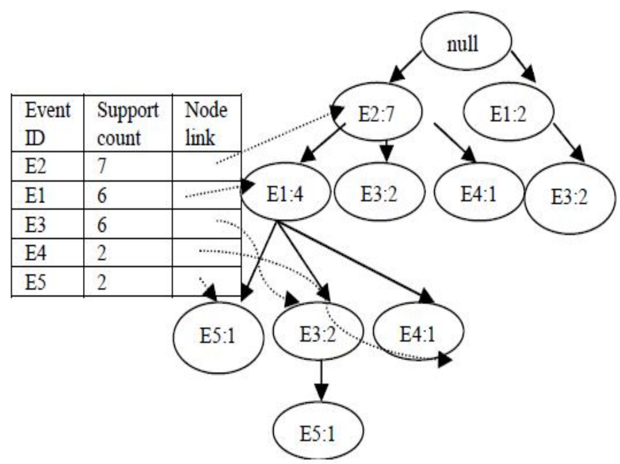

Fig. 2. FP Tree Construction

The single path generates all the combinations of frequent patterns:{E2,E5:2},{E1,E5:2},{E2,E1,E5:2}.Simil arly frequent patterns are generated for E4, E3, and E1.

TABLE II
MINING THE FP TREE BY CREATING CONDITIONAL PATTERN BASES

Event	Conditional Pattern base	Conditional FP Tree	Frequent patterns generated
E5	{{E2,E1:1} ,{E2,E1,E3:1}}	<E2:2,E1:2>	{E2,E5:2},{E1,E5:2}, {E2,E1,E5:2}
E4	{{E2,E1:1}, {E2:1}}	<E2:2>	{E2,E4:2}
E3	{{E2,E1:2},{E2:2}, {E1:2}}	<E2:4,E1:2> ,<E1:2>	{E2,E3:4}, {E1,E3:4}, {E2,E1,E3:2}
E1	{{E2:4}}	<E2:4>	{E2,E1:4}

Algorithm: FP_growth. Mine frequent itemsets using an FP tree by pattern fragment growth

Input:
A database
Min_sup, the minimum support count threshold
Output: the complete set of frequent patterns.
Method:
The FP tree is constructed.
The FP tree is mined by calling FP_growth (FP_tree, null), which is implemented as follows:
FP_growth (*Tree*, α)
For each (*ai* in the header of *Tree*) do
β := *ai* U α
Generate (β with support = *ai*.support)
Construct β's conditional base pattern
And β's conditional FP-Tree *Tree*β
If *Tree*$\beta \neq O$
Then call FP-growth (*Tree*β, β)
Initially call: FP-growth (*Tree, null)*
Where α & β are combination of the nodes in a path of the tree.

C. *Fusion Module*

Event fusion $e_i \ldots e_j$-----fuse-----$\rightarrow e_k$ It is used for merging two different events so as to track attack behavior by combining multiple alert events by using the information fusion techniques such as the D-S evidence theory so as to minimize the false positive rate. The most elementary concept of D-S evidence theory is the basic probability assignment function m defined as:

$$(P\ \{C, W\}) \longrightarrow [0, 1],$$
$$m\ (¢) = 0,$$
$$m(C) + m(W) + m(\{C, W\}) = 1\ ,$$

Where, C refers to correct alert, W refers to wrong alert m(C) refers to the confidence level of a security alert detected by a sensor.

The basic probability assignment function of combined evidence is

$$m\ (e) = K^{-1} \sum_{e_1 \cap e_2 = e} m_1(e_1) m_2(e_2)$$
$$= m_1\ (e_1) \oplus m_2\ (e_2)$$

Where, K refers to the normalization factor,

$$K = \sum_{e_1 \cap e_2 \neq ¢} m_1\ (e_1)\ m_2\ (e_2)$$

IV. SIMULATION AND EVALUATION OF RESULTS

The KCANS system developed is applied in the experiment. A wireless network of 100 nodes is created. In the simulation 5 attacker nodes are implemented to attack the sink node continuously. The simulation parameters are shown in Table III. Constant bit rate traffic is generated by using UDP agent to test the working of the system. ICMP ping sweep technique is implemented by the attacker to perform DDoS attack. The alert events generated by the intrusion detection sensors are gathered by the agents and passed to the data processing module. The knowledge discovery module invokes the FP tree algorithm to generate frequent attack patterns and then the fusion module fuses events using D-S evidence theory. Finally the security context data is passed to the security manager for decision making.

TABLE III
SIMULATION PARAMETERS

Parameters	Description
Number of nodes	100
Initial energy	100J
CST	250m
Routing	Adhoc
Topology	Flat grid
X dimension of topology	100m
Y dimension of topology	100m
Mac type	802.11
Antenna	Omni
Interface queue type	Queue/Drop Tail/PriQueue

The performance of the system is tested by varying the number of nodes from 100,120,140 and 160. From the Fig. 3 it is observed that the maximum value of average end to end delay with KD technique is .008s whereas without knowledge discovery delay reaches to a maximum value of .031s. Hence the avg. end to end delay is considerably less with knowledge discovery as compared to without knowledge discovery technique. As the number of effective alert events are greatly reduced by event filtering and fusion.

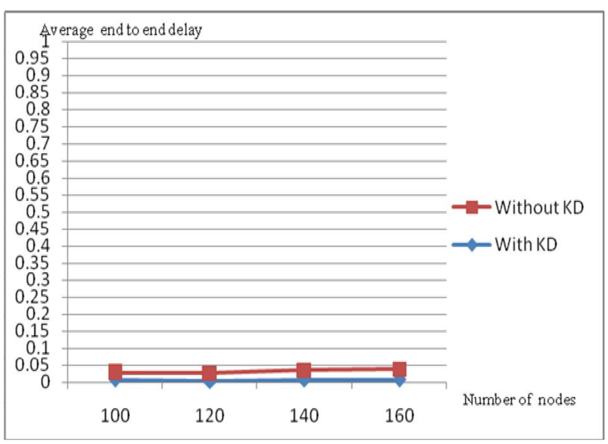

Fig. 3. Number of nodes vs Avg. end to end delay

Fig. 4 shows that .309J is the highest value of average energy consumption with KD technique and .397J is the highest value without KD. Thus the average energy consumption with KD is less as compared to without KD hence the performance of the system improves.

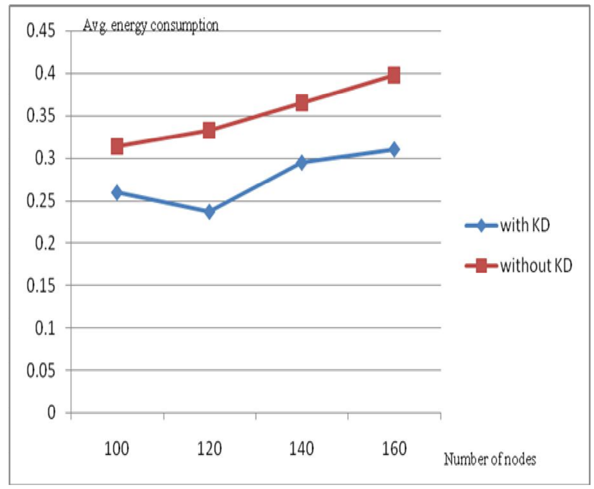

Fig. 4. Number of nodes vs Avg. energy consumption

The packet drop greatly reduces and packet delivery ratio shows a significant improvement with knowledge discovery technique as attacker is detected and prevented from sending packets. While performing the DDoS attack the attacker tries to send many packets to the same node in a short interval of time thus preventing the node from accepting legitimate traffic. Fig. 5 shows that with KD technique the packet delivery ratio reaches to 100 whereas without KD the packet delivery ratio reaches to a maximum value of 84.44.

Fig. 5. Number of nodes vs Packet delivery ratio

The average rate of successful packet delivery over the communication channel increases due to the decrease in packet loss, the throughput of the system increases. The throughput with KD technique reaches to a maximum value of 96310bps whereas without KD the throughput is 79075bps. This is shown in Fig. 6.

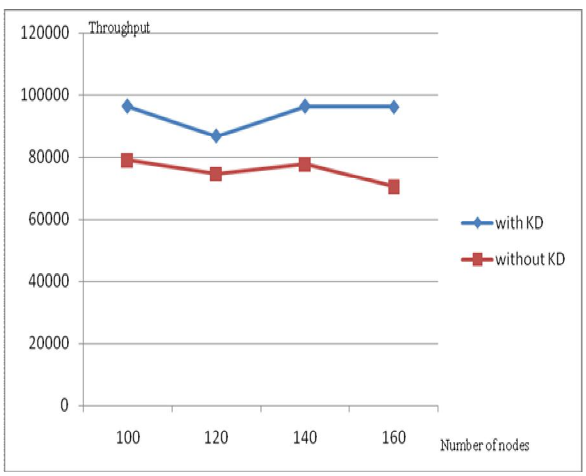

Fig. 6. Number of nodes vs Throughput

V. CONCLUSIONS & FUTURE WORK

Application of KCANS supports for exact designing and effective construction of network security context and overcomes the drawbacks of the existing security controls. The intruder is detected and deleted from the network. Using the event fusion technique many trivial attacks can be converted into DDoS attack. The number of effective alert events is greatly reduced. Security analysts do not need to face thousands of alerts that prevent them from managing and controlling their networks effectively.

The results show that the performance of the system remarkably improves with knowledge discovery technique in terms of average end to end delay, average energy consumption, packet delivery ratio and throughput.

As this work continues, the difficulties of actual forecasting of evolving fatal malicious activities are to be explored.

REFERENCES

1. J.R. Goodall, W.G. Lutters and K. Anita, “The work of intrusion detection: rethinking the role of security analysts,” in Proc. of the Tenth Americas Conf. on Information System, New York, 2004, pp. 1421-1427.

2. M.E. Hellman, W. Diffie., “An Introduction to Cryptography” volume 67, pages 397–427. Proceedings of IEEE, March 1999.

3. M. R. Endsley, “Design and evaluation for situation awareness enhancement”, Proceeding of the human factors society 32nd annual meeting, Santa Monica, CA, pp.97-101, 1988.

4. Cyril Onwubiko, “Functional Requirements of Situational Awareness in Computer Network Security”, 2009 IEEE.

5. T. Bass, "Multi sensor Data Fusion for Next Generation Distributed Intrusion Detection Systems”, Invited Paper 1999 IRIS National Symposium on Sensor and Data Fusion, pp.24-27, May 1999.

6. T. Bass, "Intrusion Detection Systems and Multisensor Data Fusion ", Communications of the ACM, Vol. 43, No. 4, April 2000.

7. Liu Xiaowu, Wang Huiqiang, Lai Jibao, and Liang Ying, “Network Security Situation Awareness Model Based on Heterogeneous Multi-sensor Data Fusion”, 2007 IEEE.

8. Liu Xiaowu, Yu Jiguo, Wang MaoLi, “Network Security Situation Generation and Evaluation Based on Heterogeneous Sensor Fusion”, 2009 IEEE

9. Ying Liang, Hui-Qiang Wang, Ji-Bao Lai,“Quantification of Network Security Situational Awareness Based on Evolutionary Neural Network”, Proceedings of the Sixth International Conference on Machine Learning and Cybernetics, Hong Kong, 19- 22 August 2007 IEEE.

10. Juan Wang, Feng-li Zhang, Jing Jin, Wei Chen, "Alert Analysis and Threat Evaluation in Network Situation Awareness", 2010 IEEE.

11. Babak Khosravifar, Jamal Bentahar, "An Experience Improving Intrusion Detection Systems False Alarm Ratio by Using Honeypot", 22nd International Conference on Advanced Information Networking and Applications, 2008 IEEE.
12. Zhao-Yang Qu, Ya-Ying Li, Peng-Li, "A Network Security Situation Evaluation Method Based on D-S Evidence Theory", 2nd Conference on Environmental Science and Information Application Technology, 2010 IEEE

13. Fang Lan, Wang Chunlei, and MaGuoqing, "A Framework for Network Security Situation Awareness Based on Knowledge Discovery" 2nd International Conference on Computer Engineering and Technology 2010 IEEE.

14. J Hall, J Pei, Y Yin, "Mining frequent patterns without candidate generation".2000 ACM, SIGMOD int'I Conf on Management of Data (SIGMOD'OO), DallaS, TX, 2000

15. Jia Han, Micheline Kamber., "Data Mining concepts and techniques", secondedition2006,ElsevierInc.