IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 10, October 2017

Towards Effective Trust-Based Packet Filtering in Collaborative Network Environments

D. Rohini[1], Dr. S. Adaekalavan[2]

Research Scholar, Department of Computer Science, J.J College of Arts and Science (Autonomous), Pudukkottai[1]

Assistant Professor, Department of Computer Science, J.J College of Arts and Science (Autonomous), Pudukkottai[2]

Abstract: Overhead network packets are a big challenge for Intrusion Detection Systems (IDSs),which may increase system burden, degrade system performance and even cause the whole system collapse, when the number of incoming packets exceeds the maximum handling capability. To address this issue packet filtration is considered as a promising solution and our previous research efforts have proven that designing a trust-based packet filter was able to refine unwanted network packets and reduce the workload in a local IDS. With the development of internet co-operation, Collaborative Intrusion Detection Environments (eg., CIDNs) has been developed, which allow IDS nodes to collect information and learn experience from others. However, it wouldnot be effective for the previously built trust-based packet filter to work in such a collaborative environment. Since the process of trust computation can be easily compromised by insider attacks. In this paper, we adopt the existing CIDN framework and aim to apply a collaborative trust-based approach to reduce unwanted packets. More specifically, we develop a collaborative trust-based packet filter which can be deployed in collaborative networks and be robust against typical inside attacks (eg., betrayal attacks). Experimental results in various simulated and practical environments demonstrate that our filter can perform effectively in reducing unwanted traffic and can defend against insider attacks through identifying in reducing unwanted traffic and can defend against insider attacks through identifying malicious nodes in a quick manner as compared to similar approaches.

Keywords: Intrusion Detection System, Network, Trust based packet filtering.

I. INTRODUCTION

Network intrusions such as worms, Trojans and DDoS attacks are a big threat for computer networks and have already become more sophisticated to detect and defend. For instance, McAfee's threat prediction report indicates that intrusions over the Internet would still be prevalent in future years. The potential damage of these intrusions could be significant if they are not detected timely. To address this problem, intrusion detection systems (IDSs) have been implemented at large with the purpose of defending against various attacks and they have become an indispensable component with respect to current defense mechanisms. These detection systems usually identify an intrusion through comparing observable behavior against suspicious patterns. In particular, based on different detection methodologies, an IDS can be typically classified as signature-based IDS and anomaly-based IDS. A signature-based IDS (or rule-based IDS) detects a potential attack by comparing incoming events with its stored signatures, where a signature is a kind of description that defines an attack or an exploit by means of expert knowledge. On the other hand, an anomalybased IDS tries to identify great deviations between current events and its pre-established normal profile. A normal profile often represents a normal action or a normal network connection through monitoring the normal behavior for a long period. In addition, based on the deployed locations and target events, an IDS can be classified as host-based IDS (HIDS) and network-based IDS (NIDS). The former like often resides on a local system and tracks changes made to important files and directories, while the latter like usually places on the network with the purpose of analyzing network traffic for malicious patterns. Traditionally, an IDS often works in isolation so that it might be easily compromised by novel threats and complicated attacks (e.g., DDoS) [10]. Thus, collaborative intrusion detection networks (CIDNs) [41] have been developed, which allow a single IDS node within this network to collect useful information and learn experience from other IDS nodes, aiming to enhance the overall detection performance. Nowadays, IDS collaboration has become an effective way to facilitate the communications between detection nodes, and identify novel and complex attacks. However, IDS may also encounter various issues in such collaborative environment. In this work, we focus on two challenges: namely, overhead network packets and effective trust computation. 1) Overhead network packets. For a network-based IDS (especially a signature-based NIDS), overhead network packets are a very challenging issue. The term 'overhead' here means that incoming packets exceed the maximum handling capability of an IDS. In a large-scale network, massive amounts of incoming network packets can quickly exhaust computer resources, greatly decrease the performance of an IDS, and even cause the paralysis of the whole system. Taking Snort as an example, it usually spends around about 30 percent of its total computational power

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 10, October 2017

in conducting signature matching between the signatures and incoming packet payloads, while its computational consumption can be significantly increased in a large-scale network environment. Typically, its computational burden is at least linear to the size of an input packet payload. Previous research reports have indicated that an IDS cannot ensure the detection performance under the high-traffic environments. In the era of big data, this challenge will become more thorny and attractive. Effective trust computation. In a CIDN, malicious nodes can greatly affect trust computation and decrease the effectiveness of packet filtration. As a result, the previously developed trust-based packet filter can perform well in a local system, but would not be effective to work in a collaborative network, since the process of trust computation can be easily compromised. In order to construct an effective trust-based packet filter, there is a need to evaluate a node's trustworthiness in a robust way, and defend against insider attacks (e.g., betrayal attacks) under a collaborative environment.

II. LITERATURE SURVEY

Wenjuan Li et al [1]. They described a CIDN is expected to have more power in detecting attacks such as denial-of-service (DoS) than a single IDS. In real deployment, they notice that each IDS has different levels of sensitivity in detecting different types of intrusions (i.e., based on their own signatures and settings). They proposed a machine learning-based approach to assign intrusion sensitivity based on expert knowledge and design a trust management model that allows each IDS to evaluate the trustworthiness of others by considering their detection sensitivities. In the evaluation, we explore the performance of our proposed approach under different attack scenarios. Therefore design a trust management model for CIDNs based on the notion of intrusion sensitivity aiming to emphasize the impact of an expert node in identifying malicious nodes. In particular, as a study, we develop an expert knowledge-based KNN classifier that can automatically assign the value of intrusion sensitivity to an IDS node. There are many possible topics in further work. Following work could include discussing the calculation of other trust types such as recommendation trust in the trust management model and verifying the impact of the intrusion sensitivity with even larger experiments.
Weizhi Meng et al [2]. The multi-touch is a distinguished feature on current smartphones and its impact on graphical password creation is an important topic in the literature. In this paper, our interest is to investigate the influence of multi-touch behaviours on users' habit in creating graphical passwords, especially on click-draw based GPs (shortly CD-GPS) on mobile devices. They develop a multi-touch enabled CD-GPS on smartphones and conduct two major experiments with a total of 90 participants. The study results indicate that participants are more likely to use multi-touch features to create their secrets, and multi-touch can make a positive impact on creating graphical passwords. . In real-world applications, we find that the process of creating graphical passwords can be different between a computer and a touchscreen smartphone. For instance, users can perform actions like multitouch on smartphones than on desktop computers Multi-touch has become a distinguishing feature for touch-enabled mobile phones. Osamah Mohammed Fadhil et al [3]. They described Intrusion detection systems are used to detect and prevent the attacks in networks and databases. Rough Set Attribute Reduction Algorithm is one of the major theories used for successfully reducing the attributes by removing redundancies. They described algorithm is used for selecting the minimal number of attributes has been from KDD data set. Moreover, a new K-Nearest Neighborhood based algorithm is proposed for classifying data set. This proposed feature selection algorithm considerably reduces the unwanted attributes or features and the classification algorithm finds the type of intrusion effectively. The proposed work selects only the significant features that have the high probability of predictive measure. With the reduced set, we have reduced the computational time. Further, the Enhanced K-NN classifier helped in achieving the greater accuracy. Hence, we computed the result in an efficient manner to prevent the attacks that improves the security. Hai Thanh Nguyen et al [4]. They described to perform an in-depth analysis of two main measures used in the filter model: the correlationfeature-selection (CFS) measure and the minimalredundancy-maximal-relevance (mRMR) measure. We showed that the measures can be fused and generalized into a generic feature-selection (GeFS) measure. The new approach is based on solved a mixed 0-1 linear programming problem (M01LP) by using the branchand-bound algorithm. In this M01LP problem, the number of constraints and variables is linear (O(n)) in the number n of full set features. Experimental results obtained over the KDD Cup'99 test data set for intrusion detection systems show that the GeFS measure removes 93% of irrelevant and redundant features from the original data set, while keeping or yielding an even better classification accuracy. Gotam Singh Lalotra et al [5]. They described an Intelligent Condition Random Field (ICRF) based Cuttlefish Feature Selection Algorithm (ICRFCFA) for effective decision making over medical datasets has been proposed. This proposed feature selection algorithm helps to improve the prediction accuracy in less time. An ICRFCFA for feature selection is employed then the selected features are involved with cuttlefish optimization approach for finalizing the necessary features for effective decision making with less computation cost. Experiments are carried out for evaluating the efficiency of the proposed feature selection algorithm using Diabetes and Heart disease datasets. They described an Intelligent CRF based cuttlefish algorithm (ICRFCFA) is proposed and implemented for effective feature selection in this paper to improve the performance of the medical expert system. The proposed algorithm is used to select the be stand the worst features, and then with the application of cuttlefish

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 10, October 2017

optimization approach for finalizing the necessary features for effective decision making. Future works in this direction could be the introduction of new rules for effective feature selection.
Ambusaidi et al [6], filter based feature selection could handle linearly and nonlinearly dependant data features. Classification is done by SVM classifier. Though ANN used to detect attacks in IDS but provide the less accuracy due to its design to solve this ICA was used to fuse the complex intrusion input and hence attain renowned characteristics (that is, self-determining components, ICs) about the original data. By the use of ICs, the intricate of the ANN structure design could be condensed. Then, the PSO was employed to optimize the structural parameters of the ANN. Adel Sabry Eesa, Zeynep Orman,, uses the cuttlefish algorithm (CFA) as a search tactic to determine the best subset of features and the decision tree (DT) classifier as a judgment on the selected features that are produced by the CFA.

III. PROBLEM DESCRIPTION

A.EXISTING SYSTEM
Traditionally, an IDS often works in isolation so that it might be easily compromised by novel threats and complicated attacks (e.g., DDoS). Thus, collaborative intrusion detection networks (CIDNs) have been developed, which allow a single IDS node within this network to collect useful information and learn experience from other IDS nodes, aiming to enhance the overall detection performance. Nowadays, IDS collaboration has become an effective way to facilitate the communications between detection nodes, and identify novel and complex attacks. However, IDS may also encounter various issues in such collaborative environment. In this work, we focus on two challenges: namely, overhead network packets and effective trust computation. 1) Overhead network packets. For a network-based IDS (especially a signature-based NIDS), overhead network packets are a very challenging issue. The term ‘overhead’ here means that incoming packets exceed the maximum handling capability of an IDS. In a large-scale network, massive amounts of incoming network packets can quickly exhaust computer resources, greatly decrease the performance of an IDS, and even cause the paralysis of the whole system. Taking Snort as an example, it usually spends around about 30 percent of its total computational power in conducting signature matching between the signatures and incoming packet payloads, while its computational consumption can be significantly increased in a large-scale network environment. Typically, its computational burden is at least linear to the size of an input packet payload. Previous research reports have indicated that an IDS cannot ensure the detection performance under the high-traffic environments. In the era of big data, this challenge will become more thorny and attractive.

a. METHODOLOGY
Mobile Ad hoc Network is a collection of wireless mobile nodes forming a network without using any existing infrastructure. MANET is a collection of mobile nodes equipped with both a wireless-transmitter and receiver that communicate with each other via bi-directional wireless links either directly or indirectly. A new intrusion detection system named Enhanced Adaptive Acknowledgement (EAACK) specially designed for MANETs. By the adoption of MRA scheme, EAACK is capable of detecting malicious nodes despite the existence of false misbehavior report and compared it against other popular mechanisms in different scenarios through simulation. The results will demonstrate positive performances against Watchdog, TWOACK and EAACK in the cases of receiver collision, limited transmission power and false misbehavior report. EAACK demonstrates higher malicious behavior detection rates in certain circumstances while does not greatly affect the network performances.

- The past decade, there has been a growing interest in wireless networks, as the cost of mobile devices such as PDAs, laptops, cellular phones, etc have reduced drastically.
- The latest trend in wireless networks is towards pervasive and ubiquitous computing - catering to both nomadic and fixed users, anytime and anywhere.
- A need for communication in several scenarios of deployment where it is not feasible to deploy fixed wireless access points due to physical constraints of the medium.

B.PROPOSED WORK
Blacklist Packet Filter.
This component is mainly responsible for refining network packets based on the IP confidence. As described in Section II, it consists of a blacklist and a look-up table. The blacklist contains all blacklisted IP addresses while the look-up table contains all IDS signatures indexed by the blacklisted IP addresses.
Trust Calculation Engine.
This component is responsible for collecting data from the blacklist packet filter, the deployed IDS and the collaboration component, computing the overall IP confidence and updating the blacklist periodically. The interactions among this engine, the local IDS and the blacklist packet filter are similar to the previous work. Differently, in a collaborative network, this engine can send out a query to other nodes for collecting the IP confidence. The collaboration component will help collect the corresponding feedback and forward data to this engine.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 10, October 2017

Collaboration Component

This component is responsible for collecting the data (e.g., IP confidence) from other nodes and forwarding the required information to the trust calculation engine. When this component receives a query from the trust calculation engine, it will help send out a request to the target node and collect the relevant feedback

Trust Computation

In this work, we develop two types of trust values: node trust and overall IP confidence. Node trust is used to evaluate the trustworthiness of a node, while overall IP confidence is used to evaluate the trustworthiness of an IP address, which can help generate a blacklist accordingly. Node trust. In a collaborative environment, this kind of trust value aims to evaluate the trustworthiness of a node. According to the basic CIDN framework the trustworthiness of a node can be calculated based on its responses to challenges. The challenges are sent out periodically by means of a random process. After receiving an answer to a challenge, a satisfaction level can be computed through identifying the gab between the received feedback and the expected answer. It is worth noting that the feedback from a node is ordered from the most recent to the oldest according to tk. As a result, the trust value of node i according to node j can be computed

Experimental Methodology

In the evaluation, we conduct three experiments to investigate the performance of our designed collaborative trust-based packet filter in various scenarios. To facilitate the comparison with related studies, we accept the assumptions in relation to honest and malicious nodes from . • In the first experiment, we deploy the packet filter into an honest environment, in which an honest IDS node always generates feedback based on its truthful judgment. This experiment also attempts to find an appropriate threshold based on the false rates of blacklist generation. • In the second experiment, we deploy the filter into a dishonest environment, where a dishonest IDS node always sends its feedback opposite to its truthful judgement. This experiment aims to evaluate the filter performance and the robustness of trust computation in a hazard scenario. In the third experiment, we investigate the practical performance of our filter in a real wireless Ad Hoc network (maintained by a company). The motivation is to explore its false rates of blacklist generation and packet filtration rate in a practical scenario.

IV. RESULT

A.ROUTING SET-UP PHASE

The operation of this phase is composed of three steps: step 1 is to divide the nodes into mutually exclusive subsets and each node joins any subset randomly, step 2 is to determine the minimal hop count for each node the last step is to construct the connected minimal hop count routing path for each node within any subset.

The detailed process is described as follows:

Step 1: Nodes are divided into mutually exclusive subsets.

Step 2: Determining the minimal hop count for each node.

Step 3: Constructing the connected minimal hop count routing path.

At the end of step 3 some nodes are scheduled in multiple sub- sets and the members in all subsets are not mutually exclusive. Furthermore, in any subset each member is guaranteed to find a connected minimal hop count routing path.

B. SIMULATION RESULTS:

The energy efficient routing algorithm using shortest path is used to tolerate the energy problem of the proposed system. To measure the network coverage entire region is divided into multiple circle-shaped small patches with radius 1 m. The coverage degree of a patch is approximated by measuring the number of nodes that cover the center of the patch. That is the patch is covered when the center of the patch is covered by at least one sensor node. In the following experiments each algorithm was executed 100 times to get more reliable results. For each experiment 10 different random network topologies are generated. The simulation results are plotted using the average values derived from these networks with a 95% confidence interval. A more efficient algorithm is signified by the lower values in algorithm delay/data transmission latency and number of active nodes as well as a higher value in packet delivery ratio. Besides, the algorithm also needs to guarantee the sufficient network coverage ratio.

Table 1: Parameters for Networks

Parameter	Value
Circle Region Radius	250m
Node Sensing Range	20m
Number of Nodes	100
Initial Energy per node	5j
Network bandwidth	1 mbps/s
Data Transmission rate	4096 bits/ s

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 10, October 2017

Performance Evaluation

In this section, we concentrate on describing our simulation environment and methodology as well as comparing perfor- mances through simulation result comparison with Watchdog, TWOACK, and EAACK schemes.

Simulation Methodologies

To better investigate the performance of EAACK under different types of attacks, we propose three scenario settings to simulate different types of misbehaviors or attacks.

Scenario 1: In this scenario, we simulated a basic packet- dropping attack. Malicious nodes simply drop all the packets that they receive. The purpose of this scenario is to test the performance of IDSs against two weaknesses of Watchdog, namely, receiver collision and limited transmission power.

Scenario 2: This scenario is designed to test IDSs' performances against false misbehavior report. In this case, malicious nodes always drop the packets that they receive and send back a false misbehavior report whenever it is possible.

Scenario 3: This scenario is used to test the IDSs' performances when the attackers are smart enough to forge acknowledgment packets and claiming positive result while, in fact, it is negative. As Watchdog is not an acknowledgment-based scheme, it is not eligible for this scenario setting.

C. SIMULATION CONFIGURATIONS

Our simulation is conducted within the Network Simulator (NS) 2.34 environment on a platform with GCC 4.3 and Ubuntu9.10. The system is running on a laptop with Core 2 Duo T7250CPU and 3-GB RAM.In order to better compare our simulation results with other research works, we adopted the default scenario settings in NS2.34. The intention is to provide more general results and makeit easier for us to compare the results. In NS 2.34, the default configuration specifies 50 nodes in a flat space with a size of670 × 670 m. The maximum hops allowed in this configuration setting are four. Both the physical layer and the 802.11 MAC layer are included in the wireless extension of NS2. The moving speed of mobile node is limited to 20 m/s and a pause time of 1000 s. User Datagram Protocol traffic with constant bit rate is implemented with a packet size of 512 B. For each scheme, we ran every network scenario three times and calculated the average performance. In order to measure and compare the performances of our proposed scheme, we continue to adopt the following two performance metrics [13].

1) Packet delivery ratio (PDR):

PDR defines the ratio of the number of packets received by the destination node to the number of packets sent by the source node.

2) Routing overhead (RO):

RO defines the ratio of theamount of routing-related transmissions [Route REQuest (RREQ), Route REPly (RREP), Route ERRor (RERR), ACK, S-ACK, and MRA]. During the simulation, the source route broadcasts an RREQ message to all the neighbors within its communication range. Upon receiving this

RREQ message, each neighbor appends their addresses to the message and broadcasts this new message to their neighbors. If any node receives the same RREQ message more than once, it ignores it. If a failed node is detected, which generally indicates a broken link in flat routing protocols like DSR, a RERR message is sent to the source node. When the RREQ message arrives to its final destination node, the destination node initiates an RREP message and sends this message back to the source node by reversing the route in the RREQ message. Regarding the digital signature schemes, we adopted an open source library named Botan [32]. This cryptography library is locally compiled with GCC 4.3. To compare performances between DSA and RSA schemes, we generated a 1024-b DSA key and a 1024-b RSA key for every node in the network. We assumed that both a public key and a private key are generated for each node and they were all distributed in advance. The typical sizes of public- and private-key files are 654 and 509 B with a 1024-b DSA key, respectively. On the other hand, the sizes of public- and private-key files for 1024-b RSA are 272 and 916 B, respectively. The signature file sizes for DSA and RSA are 89 and 131 B, respectively. In terms of computational complexity and memory consumption, we did research on popular mobile sensors. According to our research, one of themost popular sensor nodes in themarket is Tmote Sky [34]. This type of sensor is equipped with a TI MSP430F1611 8-MHz CPU and 1070 KB of memory space. We believe that this is enough for handling our simulation settings in terms of both computational power and memory space.

Table II

Scenario 1: Packet Delivery Ratio					
	Packet Filter 0%	Packet Filter 10%	Packet Filter 20%	Packet Filter 30%	Packet Filter 40%
DSR	1	0.82	0.73	0.68	0.66
WatchDog	1	0.83	0.77	0.7	0.67

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 10, October 2017

TWOACK	1	0.97	0.96	0.92	0.92
AACK	1	0.96	0.96	0.93	0.92
EAACK(DSA)	1	0.96	0.97	0.93	0.91
EAACK(RSA)	1	0.96	0.97	0.92	0.92

Scenario 1: Routing Overhead					
	Packet Filter 0%	Packet Filter 10%	Packet Filter 20%	Packet Filter 30%	Packet Filter 40%
DSR	0.02	0.023	0.023	0.022	0.02
WatchDog	0.02	0.025	0.025	0.023	0.023
TWOACK	0.18	0.4	0.43	0.42	0.51
AACK	0.03	0.23	0.32	0.33	0.39
EAACK(DSA)	0.15	0.28	0.35	0.44	0.58
EAACK(RSA)	0.16	0.3	0.37	0.47	0.61

D. PERFORMANCE EVALUATION

To provide readers with a better insight on our simulation results, detailed simulation data are presented in Table II.

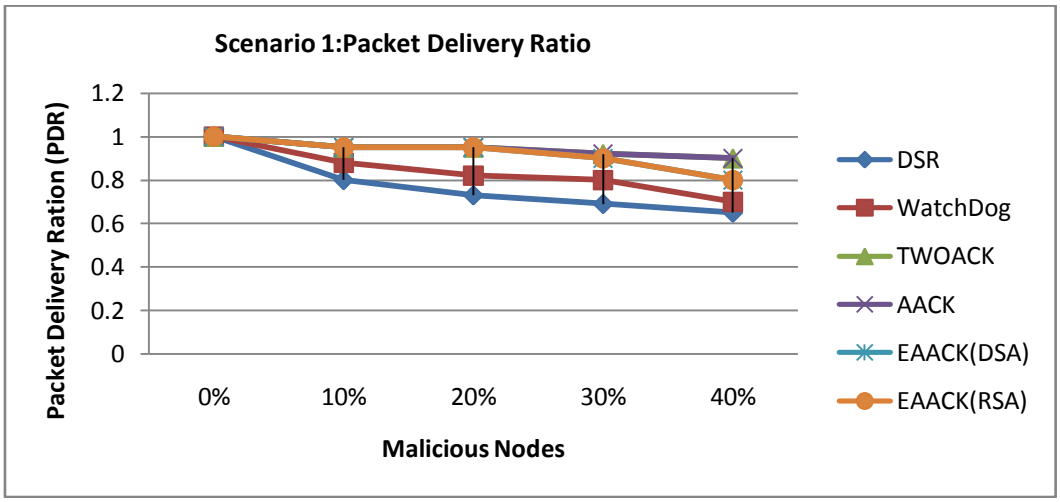

Fig. 4. Simulation results for scenario 1—PDR.

Simulation Results—Scenario 1:

In scenario 1, malicious nodes drop all the packets that pass through it. Fig. 4 shows the simulation results that are based on PDR. In Fig. 4, we observe that all acknowledgment-based IDSs perform better than the Watchdog scheme. Our proposed scheme EAACK surpassed Watchdog's performance by 21%

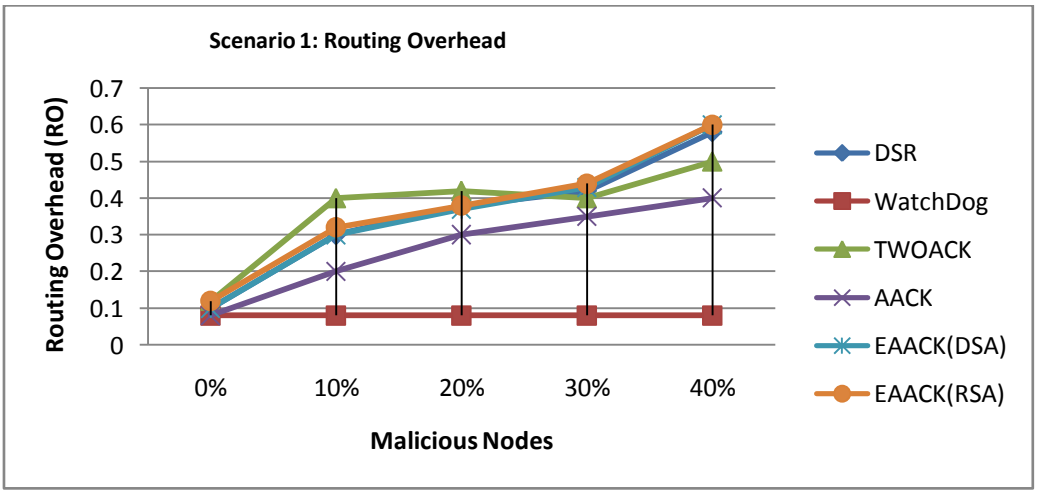

Fig. 5. Simulation results for scenario 1—RO.

E.DSA and RSA:

In scenario, we witness that the DSA scheme always produces slightly less network overhead than RSA does. This is easy to understand because the signature size of DSA is much smaller than the signature size of RSA. However, it is interesting to observe that the RO differences between RSA and DSA schemes vary with different numbers of malicious nodes. The more malicious nodes there are, the more ROs the RSA scheme produces. We assume that this is due to the fact that more malicious nodes require more acknowledgment packets, thus increasing the ratio of digital signature in the whole network overhead. With respect to this result, we find DSA as a more desirable digital signature scheme in MANETs. The reason is that data transmission in MANETs consumes the most battery power. Although the DSA scheme requires more computational power to verify than RSA, considering the tradeoff between battery power and performance, DSA is still preferable.

V. CONCLUSION

Packet-filtering has always been a major threat to the security in MANETs. In this research work, we have proposed a novel IDS named EAACK protocol specially designed for MANETs and compared it against other popular mechanisms in different scenarios through simulations. The results demonstrated positive performances against Watchdog, TWOACK, and AACK in the cases of receiver collision, limited transmission power, and false misbehavior report. Furthermore, in an effort to prevent the attackers from initiating forged acknowledgment attacks, we extended our research to incorporate digital signature in our proposed scheme. Although it generates more ROs in some cases, as demonstrated in our experiment, it can vastly improve the network's PDR when the attackers are smart enough to forge acknowledgment packets. We think that this tradeoff is worthwhile when network security is the top priority. In order to seek the optimal DSAs in MANETs, we implemented both DSA and RSA schemes in our simulation. Eventually, we arrived to the conclusion that the DSA scheme is more suitable to be implemented in MANETs. To increase the merits of our research work,

We plan to investigate the following issues in our future research:

1) possibilities of adopting hybrid cryptography techniques to further reduce the network overhead caused by digital signature;

2) examine the possibilities of adopting a key exchange mechanism to eliminate the requirement of redistributed keys;

3) testing the performance of EAACK in real network environment instead of software simulation.

Our future work will concentrate on change of classification Method in this way enhancing the proficiency of order in a diminished time. Additionally a mix of characterization systems will be used to enhance the performance

REFERENCES

[1] Deepa V. Guleria and Chavan M.K, "Intrusion Detection System Based on Conditional Random Fields", IJCSNS International Journal of Computer Science and Network Security,2013.

[2] Sannasi Ganapathy , Pandi Vijayakumar , Palanichamy Yogesh , and Arputharaj Kannan, "An Intelligent CRF Based Feature Selection for Effective Intrusion Detection", The International Arab Journal of Information Technology,2015.

[3] Osamah Mohammed Fadhil, "Fuzzy Rough Set based Feature Selection and Enhanced KNN Classifier for Intrusion Detection", Journal of Kerbala University,2016.

[4] Hai Thanh Nguyen, Katrin Franke and Slobodan Petrovi´c, "Towards a Generic Feature-Selection Measure for Intrusion Detection", International Conference on Pattern Recognition,2010.

[5] Gotam Singh Lalotra and R.S.Thakur, "An Intelligent CRF Based Cuttlefish Feature Selection Algorithm For Effective Diagnosis", International Journal of Pharmacy & Technology,2016.

[6] Yuk Ying Chung and Noorhaniza Wahid, "A hybrid network intrusion detection system using simplified swarm optimization (SSO)", Applied Soft Computing, Elsevier, vol. 12, pg. 3014-3022, 2012.

[7] Fangjun Kuang, Weihong Xu and Siyang Zhang, "A novel hybrid KPCA and SVM with GA model for intrusion detection", Applied Soft Computing, Elsevier, vol. 18, pg. 178-184, 2014.

[8] Mohammed A. Ambusaidi, Xiangjian He, Priyadarsi Nanda and Zhiuan Tan, "Building an intrusion detection system using a filter-based feature selection algorithm", IEEE Transactions on Computers, 2014.

[9] Aikaterini Mitrokotsa and Christos Dimitrakakis, "Intrusion detection in MANET using classification algorithms: The effects of cost and model selection", Ad Hoc Networks, Elsevier, vol. 11, pg. 226-237, 2013.

[10] Seung-Ho Kang and Naju, "A Feature Selection algorithm to find optimal feature subsets for Detecting DoS attacks" IEEE Conference of Decision Making, pp. 12-17, 2015.\

[11] Yang Yi, Jiansheng Wu and Wei Xu, "Incremental SVM based on reserved set for network intrusion detection", Expert Systems with Applications, Elsevier, vol. 38, pg. 7698-7707, 2011.

[12] Veronica Bolon-Canedo, Diego Fernandez-Francos, Diego Peteiro-Barral, Amparo Alonso-Betanzos, Bertha Guijarro-Berdinas and Noelia Sanchez-Marono, "A unified pipeline for online feature selection and classification", Expert Systems with Applications, Elsevier, vol. 55, pg. 532-545, 2016.

[13] Shih-Wei Lin, Kuo-Ching Ying, Chou-Yuvan Lee and Zne-Jung Lee, "An intelligent algorithm with feature selection and decision rules applied to anomaly intrusion detection" Applied Soft Computing, Elsevier, vol. 12, pg. 3285-3290, 2012.

[14] Abdulla Amin Aburomman and Mamun Bin Ibne Reaz, "A novel SVM-KNN-PSO ensemble method for intrusion system", Applied Soft Computing, Elsevier, vol. 38, pg. 360-372, 2006.

[15] Ganapathy S., Rajesh Kambattan K., Veerapandian N. and Pasupathy M, "An Intelligent Intrusion Detection System model for MANET's based on Hybrid Feature Selection", Artificial Intelligent Systems and Machine Learning, CiiT, vol. 3, pg. 13, 2011.