Sustainable Software Agent Programming

M. Prasath, B. Perumal

Abstract-Since all network vulnerabilities cannot be predicted and detected in advance and malicious intruders cannot prevent penetration into the system in any case, Intrusion Detection System (IDS) is essential to the security of a network system. Intrusion detection system technology based on mobile agents has been commonly utilized over the last several years to detect intrusion via the distributed network.Software agents are software components that run on the display device to aid, or take responsibility for, the purchase of physical information. These agents operate on the device's standard operating system and utilize low-level memory access requests from the Application Programming Interface (API) or use a specialized operating scheme for data acquisition. The system should be available and allow customized software to be executed for this strategy. A dedicated analyst interface agent presents the output of the multi-agent detection layer to the operator which retrieves more detailed information to facilitate incident analysis. Our efficiency findings demonstrate the possibility to combine high speed hardware with the sophisticated agent software based on agents

Keywords- Intrusion Detections System, Software Agents, Application Programming Interface, Software Defined Networking.

I. INTRODUCTION

Recognizing applications is essential for visibility, QoS, charging and security. SDN becomes even more essential in terms of application awareness, e.g. network virtualization, one of the most significant SDN usage cases, is benefiting from understanding of network application sort to enhance the efficiency of individual apps. Identifying the Network Traffic application name or type is a tough job. QoS marking on the IP header is not normally trusted by application programmer and is ignored by network administrators. The article introduces an effective, solid safety framework that is intended for the use on high-speed network connections for high-performance intrusion detection system [1]. We have chosen not to create a new technique of identification in our strategy, but to incorporate current techniques with a dedicated specialist's expanded confidence model. This mix enables us to correlate and compare the outcomes of the techniques used to enhance their efficiency. Therefore, IDSs are also categorized into two classifications, namely the Network Intrusion Detection System (NIDS) and the Host-based Intrusion Detection System (HIDS). Another method to rank IDSs is by how they identify intrusions. According to this ranking, there are two classifications of intrusions, namely intrusion based on anomaly and intrusion based on misuse.

Revised Manuscript Received on December 16, 2019.

M.Prasath, Pursuing, Ph.D, Computer Science and Engineering from Kalasalingam Academy of Research and Education, Srivilliputhur, India.

Dr. B. Perumal, Associate Professor in the Department of Electronics and Communication Engineering, Kalasalingam University.

Based on the deviations from the pre-established ordinary network flow models, anomaly-based IDS can recognize illegitimate data. An intelligent agent is a program module that operates continually in a specific setting that feels it and acts on the basis of external circumstances. It is therefore prepared to perform out operations in a versatile and smart way to respond to modifications in the setting. Intrusion detection systems (IDS) perform a significant part in offering network security.Agents tracking the world maintain an in-house status of previous contributions, because their activities must be linked to the old States and the new State. The goal-based agents need to be aware that the perceptions (printing of an item acquired with the eyes) do not provide sufficient data to determine the measures to be done. An outlier is an IDS that indicates an anomaly node in a network as well as the anomaly information in distributed systems. An outlier rapidly identifies an object within a malicious-intentioned scheme.

Therefore, algorithms for pre-processing information aid enhance efficiency as well as precision in identification and decrease coaching time. This job is mainly used for providing safe conditions in severe circumstances such as battlefields, earthquakes, areas and locations where natural disasters such as storms happen. The mobility of such networks is creating new vulnerabilities in a fixed wired network that don't occur. Although many Security mechanisms were previously suggested to safeguard MANETs against vulnerabilities, the implementation of fresh types of assaults is proving that most of the existing safety policies and techniques are useless in the present internet situation.

In addition, the drastic growth in Internet assaults has resulted in most delicate data being damaged. Since attacks are increasingly complex, and assailants are concentrating on new vulnerabilities, new types of attack cannot be identified or handled by the current intrusion detection methods. A smart intrusion detection method must therefore be created which can manage both known and unknown assaults.

II. RELATED WORK

The anomaly detection system produces a normal model of behaviour, which detects any variation from it, and the latter is detected using methods such as statistical analysis, machine learning, neural networks and sequence analysis as suspect abnormal behavior and differences from normal activity [2]. In order to reduce false positives while detecting anomalies, IDS architecture suggested by Nadkarni and Mishra [3], which adapts thresholds to determine malignant compliance, is based on a tracking strategy of the composite.The IDS are two types, one is the detection of abnormalities and the other is misuse.

Sun [4],who effectively identified routing disorders, proposed a routing anomaly architecture IDS classification. It is a routing table that regularly changes and detects anomalies using two parameters: (1) percentage of

tracking inputs (PCR) change, and (2) percentage of hops count changes (PCH).The IDS are traditionally classified as a network or host based system. Control and monitor network-based schemes and identity packet communications for suspect behaviour. Host-based devices for single clients, on the other side, are built and operated on low-label scheme information such as machine call trends, file entry or application use. You can monitor suspect conduct or search for prospective dangers. [5] For the spread intrusion detection, Porras and Neumann introduced the EMRALD model. It uses monitoring at customers, fields and business layers to create a structure of assessment. It is used in and between monitor systems as part of a subscription communications process.

The architecture of the intrusion detection scheme, constructed from a series of dispersed, independent and collaborative officers, was shown by Gopalakrishna and Spafford. The officials exclusively demand and obtain data according to their concerns in the interest-based collaboration and Communication model suggested by the writers.Ning et al. acknowledged the significance of aCooperative Intrusion Detection System (CISL). The writers have therefore suggested an expansion to the CISL that enables intruder tracking elements to define applications for specific data from other elements.

Farhan[6],are proposing IDS architecture centered on mobile agents to decrease a cooperative intrusion detection system in the amount of false positives. For discussing safety security vulnerabilities and for detecting assaults, SenRecommends a cluster spread IDS architecture. It utilizes a vibrant hierarchical strategy, which increasingly aggregates, analyses and reduces the amount of intrusion information gathered through nodes as they flow up the cloud top and communicates between the node members for cooperative intrusion detection.

[7] Zeng and Guo suggested IDS based on agents which can be incorporated into corporate information systems apps. There are three types of officers in the scheme: customer, server, and communications agent officers. The officers are incorporated into a device safety-enhancing access control model. [8] Da Silva, have suggested portable IDS that utilize tracking systems depending on device conduct models and wireless information streams. This system is especially suitable for wireless applications with limited resources.

[9] Teng, have established an effective algorithm for reduced attributes, making the coherent decision table simpler. They have demonstrated that a decrease of understanding is possible and efficient to reduce characteristics that allow an enormous information collection to be classified. In the literature, there are many publications which address the identification, ranking and outlier identification of intrusions. [10] Angiulli,have suggested a distance-based outlier detection process that is used to discover and submit a subset of bottom outliers in an unlabelled information collection called an outlier solution officer.

III. SOFTWARE AGENT

A software agent is a computer program acting on behalf of a user or another program in an agency relationship derived (to do) from Latin agere. Such an "Action on behalf of" requires the authority to determine, wherever possible, the appropriate action for the user. Agencies are recognized as robot bots informally. [11] They may be used as a software package such as the chat bot running on a telephone (eg. Siri) or other computer is running with a robot. It is functionality.Software agents can be independent or cooperate with other officers or individuals. Software officers that interact with individuals (e.g. chatbots, communication robots for human beings) can develop humane characteristics such as comprehension and speaking the natural language, personalities or humanoid embodiment (see Asimo). Related and derived concepts include smart agents (in particular, displaying certain aspects of artificial intelligence (such as reasoning), independent agents (able to change their objectives), distributed agents (run on physically distinct computers), multi-agent systems (distributed agents which work together to achieve an aim which cannot be active).

A self-contained software agent has basic attributes that

- Do not invoke exclusively for a job, but enable yourself,
- May be in a host's waiting position, perceiving background,
- Upon starting conditions, may run status on a host,
- Do not need user interaction,
- Other duties may be invoked including communication.

In artificial intelligence, an intelligent agent (IA) relates to an independent organization that operates, guiding its operation towards attaining objectives (i.e. being an officer), towards an setting that uses sensor reflection and consequent actuators (i.e. being smart). To accomplish their objectives, intelligent agents can also discover or use understanding. They can be very easy or complicated. An example of an intelligent agent is considered a reflex machine, such as a thermostat. Intelligent agents are often schematically defined as a computer-like abstract autonomous scheme. For this purpose, smart agents are sometimes referred to as abstract intelligent agents (AIA) to differentiate them as software structures, biological systems, or organisations from their actual reality applications.

Intelligent agents in artificial intelligence are strongly linked to economic agents, and variants of the intelligent agent paradigm are explored in cognitive science, morality, practical argument theory, as well as in many interdisciplinary socio-cognitive modelling and machine cultural models.Smart agents are also tightly associated with software agents (an independent computer program that performs functions on account of customers). In computer science, the word "smart officer" can be used to describe to a software agent with some intelligence, regardless of whether Russell and Norvig's concept are not a rational agent. For instance, "smart officers" are also called independent programs used for operator assistance or information mining (sometimes referred to as bots).

Intelligent agents were described in a variety of respects. The following characteristics should be present in IA systems:

- Adapt fresh problem-solving guidelines incrementally
- Adapt in real time online
- Can be analysed in aspects of behaviour, mistake and achievement.
- Learn and enhance by interacting with the setting (embodiment)
- Learn fast from big quantities of information

- Have memory and recovery capabilities based on memory
- Have parameters to represent memory, age, forgetting, etc. in the short and long term.

Learning has the benefit that it enables officials to originally work in unidentified settings and become more skilled than their original understanding alone could allow the most significant difference between the learning components, which were accountable for creating changes and the efficiency component which is accountable for choosing internal activities. A critical feedback on how the agent does and how the performance element should be altered to do better in the future is used in the learning element. The aspect of success is what we deemed to be the agent before: it requires perceptions into consideration and chooses behavior. The "problem generator" is the last element of the learner. It is in charge of proposing measures that contribute to fresh and informative feelings. Intelligent agents are used to interpret client's requirements for individualizing customer service as integrated internet staff.

IV. SOFTWARE AGENTS AND THEIR USE FOR SECURITY ENHANCEMENT

The following properties for software agents are [IO]:
Autonomy: They can operate without the direct intervention of humans.
Cooperativity:They can cooperate with other agents.
Reactivity: You can sense your atmosphere and react to modifications in it in a prompt manner.
Proactivity: You can identify trends in your setting by following the action, showing goal-oriented behavior.
Several security improvement approaches deploy agent technologies as an instrument for the detection of abnormal behaviour, including a key agent on a server and a user agent operating on each client workstation.
A manager and expert system is housed in the central agent. The manager always monitors each customer asking for data on current user behavior. These data are then reviewed by the specialist system, which includes rules on valid user behaviour. We constructed a smart security agent scheme with a key software agent on one computer on Windows NT, and client end software agents live in every customer workstation. The software for each agent type was written on the Microsoft Windows NT Version 4 environment of 10/100Mbps network in SUN Java JDK version 1.2.
A communication thread is a distinctive method created by the key officer to communicate information to the end-user in reaction to a signal from the end-user officer. Unique processes allow the core agent to communicate effective and efficiently with each user agent so that network monitoring can be carried out quickly. The process will be killed after the core agent reacts to a user agent. The scheme utilizes a technology for binary identification. Invalid behavior, the present behavior of the user is compared to its typical conduct, and the present behavior is compared with a collection of overall laws regulating the legitimate conduct of applications administrators. The user's historical profile contains typical conduct.
The key officer creates the historical user profile through statistical analysis of the scheme server's audit log tile with Marcov strings. There are four different components of the user agent software: a transmitter, a sensor, a readers profile and a comparator. The sensor monitors the various applications of software (for instance, a word processor, a tablet) run by the user on that workstation. When a user logs into a sensor every 5 seconds the user activity is monitored and user identification and process identification of each application is recorded.
The receiver provides this data to the key officer after the first survey by the detector. We have killed the core agent process as part of the experiment. The effect was that new users posed a high safety risk as either the user's historical profile or the behavior rules of the system administrator were not downloaded from the user agent.

V. CONCLUSION

In each customer workstation we learned how to enhance safety through the use of software agents where a key software representative lives on one server in a Windows NT network scheme and where customer end officers live. The importance of our strategy is to take all choices and measures regarding incorrect customer conduct. This implies that safety breaches in the customer workstations can still be identified and suitable measures can still be done even if the key officer method is no longer functioning.

REFERENCES

[1] XianFeng, Du, and Qiang ZanXia. "A model of intrusion detection system based on aglet with multi-agent." In *2010 International Conference on Computer Application and System Modeling (ICCASM 2010)*, vol. 6, pp. V6-232. IEEE, 2010.
[2] Huang, Weijian, Yan An, and Wei Du. "A multi-agent-based distributed intrusion detection system." In *2010 3rd international conference on advanced computer theory and engineering (ICACTE)*, vol. 3, pp. V3-141. IEEE, 2010.
[3] Nadkarni, Ketan, and Amitabh Mishra. "A novel intrusion detection approach for wireless ad hoc networks." In *2004 IEEE Wireless Communications and Networking Conference (IEEE Cat. No. 04TH8733)*, vol. 2, pp. 831-836. IEEE, 2004.
[4] Sen, Jaydip. "An intrusion detection architecture for clustered wireless ad hoc networks." In *2010 2nd International Conference on Computational Intelligence, Communication Systems and Networks*, pp. 202-207. IEEE, 2010.
[5] Bose, S., S. Bharathimurugan, and A. Kannan. "Multi-layer integrated anomaly intrusion detection system for mobile adhoc networks." In *2007 International Conference on Signal Processing, Communications and Networking*, pp. 360-365. IEEE, 2007.
[6] Farhan, A. F., D. Zulkhairi, and M. T. Hatim. "Mobile agent intrusion detection system for mobile ad hoc networks: A non-overlapping zone approach." In *2008 4th IEEE/IFIP International Conference on Central Asia on Internet*, pp. 1-5. IEEE, 2008.
[7] Zeng, Xiang, Rajive Bagrodia, and Mario Gerla. "GloMoSim: a library for parallel simulation of large-scale wireless networks." In *Proceedings. Twelfth Workshop on Parallel and Distributed Simulation PADS'98 (Cat. No. 98TB100233)*, pp. 154-161. IEEE, 1998.
[8] Hegazy, Islam M., Taha Al-Arif, Zaki T. Fayed, and Hossam M. Faheem. "A multi-agent based system for intrusion detection." *IEEE Potentials* 22, no. 4 (2003): 28-31.
[9] Balasubramaniyan, Jai Sundar, Jose Omar Garcia-Fernandez, David Isacoff, Eugene Spafford, and Diego Zamboni. "An architecture for intrusion detection using autonomous agents." In *Proceedings 14th annual computer security applications conference (Cat. No. 98EX217)*, pp. 13-24. IEEE, 1998.
[10] Mukkamala, Srinivas, and Andrew H. Sung. "Detecting denial of service attacks using support vector machines." In *The 12th IEEE International Conference on Fuzzy Systems, 2003. FUZZ'03.*, vol. 2, pp. 1231-1236. IEEE, 2003.
[11] Farid, Dewan, Jerome Darmont, Nouria Harbi, Huu Hoa Nguyen, and Mohammad Zahidur Rahman. "Adaptive network intrusion detection learning: attribute selection and classification." 2009.

AUTHORS PROFILE

M.Prasath was born at Madurai, India in 1990. He graduated in Computer Science and Engineering from Anna University affiliated college and post graduated in Computer Science and Engineering from Kalasalingam Academy of Research and Education, Srivilliputhur, India. He is pursuing his PhD in Computer Science and Engineering (Intrusion Detection System in Software Defined Networking) from Kalasalingam Academy of Research and Education. His research interest includes Network Security, Deep Learning Algorithms, Machine Learning Algorithms and Software Agents. He has published 1IEEE International Journals, published 1 paper in International Level Conference.

Dr. B. Perumal was born at Bodinayakanur, India in 1980. He graduated in Electronics and Communication Engineering from Madurai Kamaraj University and post graduated in Digital Communication and Network Engineering in 2006 from Anna University, Chennai, India. He has done his PhD (Medical Image Compression) in 2016 from Kalasalingam University, Krishnankoil. He has done his Post Doctoral Fellowship in 2018 from Cracow University of Technology, Poland. Now he is currently working as Associate Professor in the Department of Electronics and Communication Engineering, Kalasalingam University. He has published 10 International Journals, published 19 papers in International Level Conferences and 15 National level Conferences. His research interests include Mobile Computing, Wireless Sensor Networks, Cloud computing and Bio-medical Instrumentation and Medical Image Compression.