Cooperative Machine Learning For Intrusion Detection System

Ravindra Bhosale, Sandip Mahajan, Dr. Parag Kulkarni

Abstract— Firewall is employed for defense however they are doing not offer full protection. This ends up in implementation of intrusion detection and interference system. Intrusion detection system collects the knowledge from system event, system logs, or from network packets. Implementing intrusion detection system victimization single agent is easy however its simplicity ends up in degraded performance, as single agent might not ready to handle every event to supply correct result on time i.e. may not deliver the goods real time demand of the network. Also number of attacks and vulnerability areas are rising. This problem ends up in implementation of intrusion detection system using multi-agent approach wherever single agent works for particular operation. During this new projected multi-agent intrusion detection system the agent posses few data with him. Using influence diagram each agent generates the local decision and learns from the choice that updated in local information. Each agent sends his call to al l alternative agent. Victimization this as further call with native calls every agent tries to enhance his call capability..

Index Terms—Multi-agent, Machine Learning, Cooperative learning, Intrusion Detection System (IDS), reinforcement learning, influence diagram.

1 Introduction

Use of internet in day to day life has increases security problem. Still the date numbers of methods like use of anti-virus, Firewall, intrusion detection system (IDS) are used to provide the security in computer. In March 2012 computer security labs in Iran, Russia and other announced appearance of Flame malware and stated Flame as most complex malware ever found. Flame had impacted thousands of computer but not a single anti-virus or _rewall was able to identify that. This indicates firewall, antivirus and simple IDS security methods are not so useful for such such uncertain environment or unknown attack without modifying them. This problem can be minimized by taking decision in uncertain environment. Influence diagram is able to take decision when environment is uncertain. For high performance and real-time protection with no network latencies or overloading multiple agents are required, where multiple agents can simultaneously use incomplete information with their database to take local decision based on their learning.

1.1 Intelligent agent

An agent is anything that has ability to perceiving its environment through sensors and acting upon that environment through actuators. The term percept to refer to the agent's perceptual inputs at any given instant. An agent's percept sequence is the complete history of everything the agent has ever perceived from its environment. An agent should select an action that is expected to maximize its performance.

An intelligent agent is one that is capable of flexible autonomous action in order to meet its design objectives where flexibility means reactivity, proactiveness and social ability. Reactivity property indicates that for satisfying design goal an agent must respond in a timely fashion. Proactiveness indicates says intelligent agents must be capable of taking the initiative to satisfy its design goal. Whereas social ability indicates an intelligent agent are capable of interacting with other agents which may be another software agent or human being in order to achieve its design goal [14].

According to Wooldridge there are number of features which appreciate use of agent technology like[9]

- The environment is open, or at least highly dynamic, uncertain, or complex.
- Cooperative property of agent can solve complex problem.
- Centralized solution due to Distribution of data, control is extremely difficult or at worst impossible.
- Legacy systems. That is, software technologically obsolete but functionally essential to an organization.

1.2 Multi-agent system

Multi-agent System (MAS) Multi-agent systems are computational systems in which two or more agents interact or work together to perform some set of tasks or to satisfy some set of goals. MAS possess following properties:

- No global system control
- Decentralized and incomplete information
- Asynchronous computation

MAS systems may be comprised of homogeneous or heterogeneous agents. An agent in the system is considered a locus of problem-solving activity, it operates asynchronously with respect to other agents, and it has a certain level of autonomy. Agent autonomy relates to an agents ability to make its own decisions about what activities to do, when to do them, what type of information should be communicated and to whom, and how to assimilate the information received. Autonomy can be limited by policies built into the agent by its designer.

Cooperative agents work toward achieving some common goals, whereas self-interested agents have distinct goals but may still interact to advance their own goals. In the latter case, self-interested agents may, by exchanging favours or currency, coordinate with other agents in order to get those agents to perform activities that assist in the achievement of their own objectives.

Scientific research and practice in multi-agent systems, which in the past has been called Distributed Artificial Intelligence (DAI), focuses on the development of computational principles and models for constructing, describing, implementing and analyzing the patterns of interaction and coordination in both large and small agent societies. Multi-agent systems research brings together a diverse set of research disciplines and thus there is a wide range of ideas currently being explored [4][7][9][11][12].

1.3 Influence Diagram

Influence diagrams are a powerful graphical representation for decision models, complementary to decision trees. Influence diagrams and decision trees are different graphical representations for the same underlying mathematical model and operations. Influence diagrams represent the probabilistic structure of complex problems compactly, facilitate communication between analysts and decision makers, and form the basis for efficient and easy-to-use computer-based tools.

The arcs represent relationships between the nodes. A decision node (drawn as a rectangle) provides the decision alternatives under consideration. A chance node (drawn as a circle or oval) represents a variable whose value is a probabilistic function. An arc between two chance nodes indicates that a probabilistic relationship between the two events might exist. A probabilistic relationship exists when the occurrence of one of the events affects the probability of the occurrence of the other event.

2 LITERATURE REVIEW

Any Organizations are striving to maintain confidentiality, integrity and availability of their networkedresources and a number of techniques have been employed to guard against network intrusion. However, even though these measures provide a level of security, they have been found to be lacking in a number of ways.

Many methods for detecting malicious intruders (likefirewalls, password protected systems) currently exist. Afirewall is a hardware or software solutions used to enforce security policy on a private network. It is mostlyused to control traffic to or from a private network. However, these are but just a list of permit and deny rules; therefore they may not always have the ability to detectintrusions.

Cryptography hides information from unauthorized users, however this method makes it hard toknow whether any attack has taken place.So, and these traditional methods are becoming increasingly vulnerable and inefficient due to their inherentproblems. As a result, new methods for intrusion detection that are not hampered by vulnerability and inefficiency must be developed.

2.1 Intrusion Detection System

Intrusion Detection Systems look for attack signatures, which are specific, patterns that usually indicatemalicious or suspicious intent. When the IDS look forthese patterns in network traffic via a promiscuous interface it is considered a Network Based IDS. Traditional systems in place for intrusion detection primarily use a method known as finger printing to identifymalicious users. Fingerprinting requires the compilationof the unique traits of every type of attack on a computer system. Each generated fingerprint is first added to the attack database of a detection system and then compared to all subsequent user connections for classification as either a malicious or normal connection. This trait compilation is typically accomplished through human analysis by the creators of the system. The resulting fingerprint updates must then be manually installed on each individual system in use.

An Intrusion Detection System must first be able to detect malicious user connections, for which it must have a generalized model of user behavior for comparison to users of a system. The most efficient method for generating a user model is to apply a data analysis algorithm to given training data, which is representative of real world data, and then generate an empirical model of either type of user based on this training data. When the IDS look for these patterns in network traffic via a promiscuous interface it is considered a Network Based IDS. There are three forms of a Host based IDS. Of the two main ones, the first examines the logs of the host looking for attack patterns; the second examines patterns in the network traffic (this is not done in promiscuous mode like the Network IDS). The third one is a solution that executes both Log based and Stack-Based IDS. The actual demands of effectiveness and complexity have caused the development of new computing paradigms. Intelligent agent is one of them. An agent is one who perceives its environment through sensors and acts upon that environment through actuators.

3 SYSTEM ARCHITECTURE

"Cooperativeness is not so much learning how to getalong with oth-

ers as taking the kinks out of ourselves, so that others can get along with us." Thomas S. Monson

Implementing intrusion detection system using single agent is simple but its simplicity results in degradedperformance, as single agent may not able to handle eachand every event to produce correct result on time i.e.may not achieve real time requirement of the network.Also the number of attacks and vulnerability are rising.

This problem leads to implementation of intrusion detection system using multi-agent approach where singleagent works for particular operation or function.

One of the central problems for IDSs is to build effective behavior models or patterns to distinguish normal behaviors from abnormal behaviors' by observingcollected audit data. To solve this problem, earlier IDSsusually rely on security experts to analyze the audit dataand construct intrusion detection rules manually. However, since the amount of audit data, including networktraffic, process execution traces and user command data, etc., increases vary fast, it has become a time-consuming, tedious and even impossible work for human expert's toanalyze and extracts attack signatures or detection rules from dynamic, huge volumes of audit data. Furthermore, detection rules constructed by human experts are usually based on fixed features or signatures of existing attacks, so it will be very difficult for these rules to detectdeformed or even completely new attacks. Due to theabove deficiencies of IDSs based on human experts, intrusion detection techniques using machine learning have attracted more and more interests in recent years. Machine learning is a field of study which provides the computers with the ability of learning from previous experience.

Machine learning is based heavily on statistical analysis of data and some algorithms can use patterns found in previous data to make decisions about new data.

In this new proposed multi-agent intrusion detection system the agent possessfew information with him. Using influence diagram every agent generate the local decision and learn from the decision which updated in local Database. Every agent sends his information (decision + event information) to all other agent. Using this as additional information with local information every agent tries to improve its decision capability. Following section explain the proposed agent architecture with proposed

Multi-agent system

3.1 Proposed agent architecture

Following figure shows a typical single agent of proposed multi-agent IDS system. System contains more than one such agent.

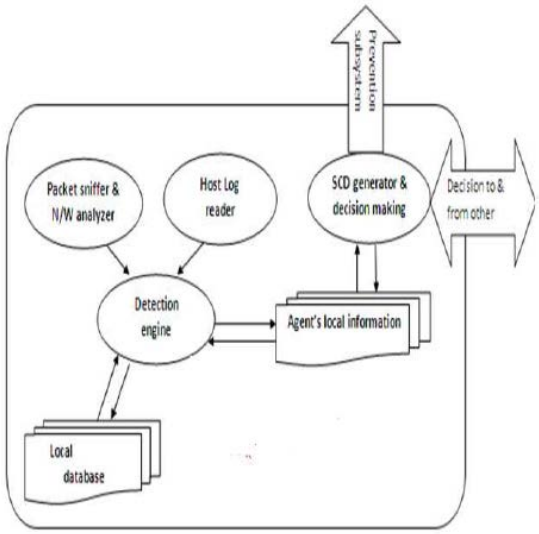

Fig 1: Proposed agent architecture

- *Packet Sniffer and analyzer:* One of the input to the proposed system is data from network. To collect network data from given network packet analyzer is used. Packet analyzer continuously monitors/ analyzes the network. Typically it is used to check packet content, source IP address of packet, packet checksum, and acknowledgement number of packet.
- *Host log reader* Operating system typically monitors different event and store this information in Log files. To detect Host based intrusion system log need to be monitor. The host log reader is used to read the system's data from its host file.
- *Local database:* This contains information possessed by an agent. This is may be different for different.
- *Detection engine:* Detection subsystem compares input data with predefined signature database and rules like sniffer rule. If match found then this information is forwarded to agent's local information. At the same time if any changes are to made in local database then local database is updated using simple communication.
- *Decision Making:* Using given formation from detection engine and the local available information, Influence diagram is created. These IFD is used to take local decision. This local decision is sends to other agent. The decision is updated when other agent sends their local decision. Based on the decision necessary signals are activated. If decision indicates intrusion then prevention subsystem is activated.

3.2 Proposed system

The figure 2 shows proposed multi-agent IDS system.In this system we have considered three hosts which canbe on same machine or different machine. Each hostagent is nothing but a single agent shown in previously.This shows typical input for each host which will benetwork connection and system log file.

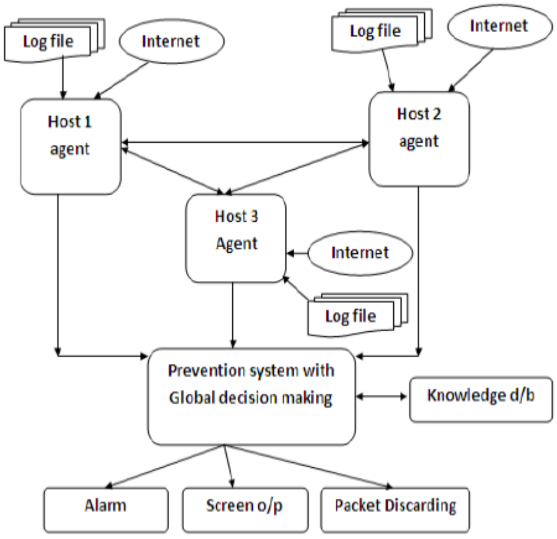

Fig 2: Proposed system architecture

3.3 System flow

- Initially every host agent consist few informationwith it.
- Every agent scan packet from its network connectionfor its intended functionality. At the same time theyalso scan host log file.
- Input data is compared with standard dataset usingdetection engine of every agent.
- Depends upon output of detection engine and previous database it creates influence diagram.
- Based upon influence diagram agent will takes itslocal decision, to improve decision capability samedecision will be forwarded to all other agent.
- When decision of other agents are received by current agent it takes new decision by considering decision of all other agent and based on that final decision about intrusion is taken.
- Final decision is then given to prevention subsystem, which takes necessary action and display the sameaction on screen.

4 DETAIL DESIGN

4.1 Mathematical model

The system to be analyzed can be in two possible states. With an intrusion (I) or without it (NI). The prior probability is representedby p. The estimation of prior probabilities is familiar to Bayesian statistics. An IDS can launch an alarm (A) or not (NA). The ROC parameters are: the probability of an alarm given an intrusion, $P(A|I) = H$ and the probability of an alarm given no intrusion, $P(A|NI) = F$.

There are three probabilities specified in the tree:

- $p1$: The probability that the detector reports no alarm.
- $p2$: the conditional probability of no intrusion given that the detector reports no alarm.
- $p3$: the conditional probability of no intrusion given that the detector reports an alarm taking the sum of products of probabilities and costs forall of the node's branches.

The expected cost at a decision node is the lowest expected cost from among thenode's outgoing branches. An operation point is definedby a pair (F,H). The probabilities of the detectors reportare calculated by applying the formula of total probability: [15]

$p1 = P(NA) = P(NA|NI).P(NI) + P(NA|I).P(I)$ (1)

$1 - p1 = P(A) = P(A|NI).P(NI) + P(A|I).P(I)$ (2)

The probabilities of the system's state depending on thedetector's rate are calculated by applying Bayes Theorem

The conditional probability of no intrusion occurs andthe detector reports no alarm will be (True negative)

$p_2 = P(NI \mid NA) = (P(NA|NI). P(NI)) / P(NA)$ (3)

Similarly from above formula we can find conditionalprobability of intrusion occurs but system falsely giveno intrusion indication will be (False negative)

$1\text{-}p_2=P(I|NA)=(P(NA|I).P(I)) / P(NA)$ (4)

The conditional probability of no intrusion occur but detector reports an alarm will be (False positive)

$p_3 = P(NI \mid A) = (P(A/NI). P(NI)) / P(A)$ (5)

And lastly there is intrusion and system report no alarmwill be (True positive)

$1\text{-}p_3 = P(I \mid A) = (P(A/I). P(I)) / P(A)$ (6)

4.2 Decision making algorithm

The utility theory and Bayesian network theory can be combined in a graphical representation, influence diagrams. An influence diagram (ID) is a compact representation emphasizing features of decision problems. The inference diagram formalism integrates the two components of knowledge, about beliefs and about actions. Influence diagrams are directed acyclic graphs with tree types of nodes decision nodes, chance nodes, and a value node. Decision nodes, shown as squares, represent choices available to the decision-maker. Chance nodes, shown as circles, represent random variables (or uncertain quantities) the same as for Bayesian networks. Finally, the value node, shown as a diamond, represents the objective (or utility) to be maximized. Decision making algorithm:

Step 1: For current environment set evidence variable as current state of agent

Step 2: For each possible value of the decision node:

• Set the decision node to that value.

• Calculate the posterior probabilities for the parent nodes of the utility node

• Calculate resulting utility function for the action

Step 3: Return the action with the highest utility.

4.3 Learning algorithm & credit assignment

Agent learning is divided into supervised, unsupervised, and reward-based learning. These methodsare distinguished by what kind of feedback the criticprovides to the learner. In supervised learning, the criticprovides the correct output. In unsupervised learning, no feedback is provided at all. In reward-based learning, the critic provides a quality assessment known as reward.

Team learning is an easy approach to multi-agent learning because its single learned can use standard single agent machine learning techniques. This sidesteps the difficulties arising from the co-adaptation of several learners that we will later encounter in concurrent learning approaches. Another advantage of a singlelearner is that it is concerned with the performance of the entire team, and not with that of individual agents.

Markov Decision Processes (MDPs) are a mathematical framework for moeling sequential decision problems under uncertainty as well as Reinforcement Learning problems. Usually, the term MDP refers to first order Markovian processes in which the current state only depends on the last previous state.

Decision process satises the Markovian property. given the current state and a constant number n of previous states, the probability of transitioning to the next state is conditionally independent of any other previous states. Multi-agent decision process can be represented by a tuple {N, S, A, O, p_t, p_o, Θ, R, B} where:

N is a set of agents

- S = {s_1, s_2, s_3…} is a set of global states.
- A_i = {a_{i1}, a_{i2}, a_{i3}…} is a set of local actions available to agent i.
- A = {A_1, A_2, A_3…} is a set of joint actions with A = A_1 × A_2 × … ×A_n.
- O_i = {o_{i1}, o_{i2}, o_{i3}…} is a set of local observations available with agent i.
- O= {O_1, O_2, O_3…} is a set of joint observations with O = O_1 × O_2 × … × O_n.
- p_t : S × S × A -> [0; 1] is the joint transition probability function from state sp to sq when taking action a_k in state.
- p_o : S × O -> [0; 1] is the observation probability function which defines the probability of observing making observations in a given state.
- Θ : O -> S is a mapping from joint observations to global states.
- R : S × A × S -> R is the reward for taking action ak in a state s_p and transitioning to s_q.
- b = {b_1,b_2,b_3…} is the vector of local belief states with b Є B, the set of joint belief states.

Here reward matrix is not agent specific but rather shared between all agents in the system. As a result the multi-agent decision process is a cooperative setting where agents have identical interests and benefit equally from choosing a certain action.

A best-response equilibrium or Nash equilibrium (NE) is a collection of policies for all agents such that π_{-I} Є $BRi(\pi_{-i})$ that is no agent can increase its reward by changing its policy given that other agents are using NE policies.

5 EXPECTED RESULT DISCUSSION

The input for IDS is nothing but 42 features from network data like network connection duration, service, number of bytes etc. Each connection can be categorized into two main class normal class and anomaly classes include DOS, Probe, U2r, r2l.

Numbers of tools are available to analyses the network like tcpdump. These tool analyses network behavior, performance and applications that generate or receive network traffic. It can also be used for analyzing the network infrastructure itself by determining whether all necessary routing is occurring properly, allowing the user to further isolate the source of a problem.It is also possible to use them for the specific purpose of intercepting and displaying the communications of another user or computer.

DARPA KDD99 dataset is simple and for real network this dataset is insufficient. So we are considering a small home network for our experiment. DARPA KDD99 dataset for each network connection that have values divided into 3 categories. First is a basic feature of network connection, which includes the duration, prototype, service, number of bytes from source IP addresses or from destination IP addresses, and some flags in TCP connections. Second is composed of the content features of network connections and the third is composed of the statistical features that are computed either by a time window or a window of certain kind of connections.

In order to have a good performance the communication must be efficient as well as IDS should be able to correctly differentiate between intrusions and valid action in a system environment. Features like false positive, true positive, false negative and true negative rateare used to measure performance of an IDS system. Proposed IDS is expected to produce high true positive rate and low false positive rate.

6 CONCLUSION

New proposed mechanism for detecting intrusions formed by an intelligent, distributed architecture based on multi-agent aspect will allows quick response against the attack complex, assesses the state of the flow captured by reference to rules and procedures. Use of Reinforcement Learning for intrusion detection helps to detect unknown attack by motivating comparator with the rewards to identify correct log files for confirmation of attack. It exemplifies the benefits of integrating various artificial intelligence techniques with the Intrusion

Detection Systems. It also provides scope for advancementsin efficient pattern matching algorithms for accurate results. Implementation of proposed system will be bestway to understand real-time issues that are not possibleto realize during designing phase of the proposed system.Looking ahead on adapting the behaviour of agentsto automate the generation mechanism of a rule whichwill corresponds to an unknown attack. This process allows the automatic feeding of the basic rules and newrules on security procedures dedicated to the recognitionof an intrusion or unknown attack.

REFERENCES

[1] AdhityaChittur,*"Model Generation for an Intrusion Detection System Using Genetic Algorithms"*.

[2] Brian Laing,JimmyAlderson,JohnRezabek,NickBond*"Intrusion detection system"*.

[3] BhagyashreeDeokar, AmbarishHazarnis,*"Intrusion Detection System using Log Files and Reinforcement Learning"*,International Journal of Computer Applications Volume 45 No.19, May 2012.

[4] Mark Adler, Edmund Durfee, Michael Huhns,William Punch,andEvangelosSimoudis,*"AAAIWorkshop on Cooperation Among Heterogeneous Intelligent Agents"*,AIMagazine Volume 13 Number 2,1992.

[5] *"Intrusion detection system"*,Information assurance tool report, 2009.

[6] HakanAlbag,*"Network and agent based Intrusion Detection Systems"*

[7] O. Oriola,*"Distributed Intrusion Detection System Using P2P Agent Mining Scheme"*,African Journal of Computing & ICT Volume 5 No. 2, March, 2012

[8] FeratSahin,*"A Bayesian Network Approach tothe Self-organization and Learning in IntelligentAgents"*

[9] Gerhard Weiss,*"Multi-agent Systems: A ModernApproach to Distributed Modern Approach to Artificial Intelligence"*,MIT Press.

[10] Eduardo Alonso, Mark daĂŹInverno, Daniel Kudenko, Michael Luck, and Jason Noble,*"Learning inMulti-agent System"*.

[11] LiviuPanait and Sean Luke,*"Cooperative Multi-Agent Learning:The State of the Art"*.

[12] David Vengerov,*"Multi-agent learning and coordination algorithms for distributed dynamic resource allocation"*.

[13] Mohammad SazzadulHoque,Md. Abdul Mukit,Md.Abu NaserBikas,*"An implementation of intrusiondetection system using genetic algorithm"*.

[14] Stuart Russel, Peter Norving,*"Artificial intelligence: A Modern Approach", Second edition*.

[15] Agust´ın Orfila , Javier Carbo,*"Intrusion detection effectiveness improvement by a multiagent system"*, International Journal of Computer Science & Applications,Volume 2 No.1,2005.