Journal of Network and Computer Applications 35 (2012) 1151–1161

Contents lists available at SciVerse ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

Review

Multi-agent systems for protecting critical infrastructures: A survey

Zubair A. Baig *

Department of Computer Engineering, King Fahd University of Petroleum & Minerals, Dhahran, Saudi Arabia

ARTICLE INFO

Article history:
Received 9 August 2011
Received in revised form
31 October 2011
Accepted 19 January 2012
Available online 2 February 2012

Keywords:
Multi-agent systems
Critical infrastructures
Security

ABSTRACT

Multi-agent systems have emerged as a very significant platform in provisioning distributed and collaborative services to critical applications. Such applications require ubiquitous agent presence in the environment for monitoring, collecting data, communication, and subsequent data analysis, where the sensitivity of the application's nature cannot be understated. Recent advances in the field of autonomous, ubiquitous, intelligent and distributed computing have led to corresponding developments in the use of collaborating multi-agents to protect critical infrastructures. Such systems have witnessed crucial demand for deployment in diverse application scenarios such as E-commerce, E-health, Network Intrusion Detection, Telematics and Transport Systems, Environmental Monitoring, as well as for distributed information processing in general. Critical infrastructures have longed for a distributed system in place for their uninterrupted and accurate operations. Multi-agents have provided one such approach towards addressing the issue of protecting such infrastructures through collaborative and distributed information processing. In this paper, a state-of-the-art on the use of multi-agent based systems for protecting five most common critical infrastructures, is presented.

1. Introduction

An agent is a computing system capable of performing an action independently without the need for support from other entities (including other agents) (Wooldridge, 2002). It may simply be a piece of software, or a piece of dedicated hardware designed for accomplishing a given task. A multi-agent system is defined as a system constituted of several collaborating agents, with each one responsible for a specific task, such as network communication, data filtering or data analysis. The pervasive nature of such systems ascertains that tasks not accomplishable by a single agent acting on its own, are assessed and addressed, to achieve a desired objective, through regular interaction between multiple distributed agents. Such an objective may include the protection of an international border against trespassing, timely detection of a bushfire, accurate analysis of the traffic state of a large metropolis, remote monitoring of a vehicle carrying critical goods from source to destination, and so forth. The resulting outcome of such collaboration between multiple agents is the aggregate property of a system as a whole, and not of a single agent (Gligor, 2004). The above property of such a system enables holistic visualization of parameters acquired by the system from multiple agents, and also ensures a high degree of accuracy in the decision making process as a response to inputs received from the agents to construe a given application scenario. A multi-agent system has applications in several fields such as: E-commerce, real-time monitoring of telecommunication networks, information handling on the Internet, air traffic control, meeting scheduling, Network Intrusion Detection (Weiss, 1999), etc. Contrary to a standard multi-agent system, an intelligent multi-agent system ought to be affected by existing environmental factors that help achieve desirable goals more effectively and efficiently. In other words, some optimization criteria is applied by an intelligent agent to accomplish underlying objectives (Weiss, 1999). Such criteria may include the application of intelligent techniques such as Neural Networks, Fuzzy Logic, Evolutionary Computing, etc., for providing enhanced responses to limited input data that the system may be subjected to, from individual agents.

Through a multi-agent based approach to provide a holistic and reliable perception of a given critical application scenario or a system state, delivery of necessary goods or services to counter a detected problem are realizable both effectively as well as in a timely manner. In addition, the use of multiple agents ascertains that the failure of a single agent will not catastrophically affect the operations of the entire system. In this paper, state-of-the-art on the use of multi-agent systems for protecting critical infrastructures is presented. In particular, classification of critical applications into five most common categories, namely, E-commerce, E-health, Intrusion Detection Systems, Telematics and Transport Systems, and Environmental Monitoring, is defined. Contemporary approaches for protecting the above-mentioned critical infrastructures based on the coordinated and collaborative operation of

* Tel.: +966 5 44017080.
E-mail address: zbaig@kfupm.edu.sa

1084-8045/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jnca.2012.01.006

multiple agents is presented. With sophisticated adversaries omnipresent in today's computer networks, it has become crucial to protect important resources and services from any attempts to malign the network, such as malicious attacks against the integrity, availability or the confidentiality. In addition, inadvertent misuse of single-agent systems may cause sub-par performances, and limited accuracy in provisioning timely services based on agent inputs. Since multi-agents have been used in a number of useful applications, I review the use of such systems for provisioning security to some of the most common applications, namely, E-commerce, E-health, Intrusion Detection Systems, Telematics and Transport Systems, and Environmental Monitoring.

The study of *Resource Protection* is associated with protecting the storage, retrieval and communication of data, as well as for protection of other physical resources such as borders. In this paper, a study of multi-agent based techniques for information and infrastructure protection under the following classes—E-commerce, Telematics, E-health, as example areas, is provided.

On the other hand, *Network security and protection* refers to the field of study of protecting one's computer network from malicious attempts to infiltrate and cause damage to it by the adversary class. The most popular forms of network security solutions are provisioned through the use of firewalls, Network Intrusion Detection Systems, and well-configured (hardened) network routers. As part of this survey, I provide an analysis of multi-agent based Network Intrusion Detection Systems.

The organization of this paper is as follows: In Section 2, I survey the use of multi-agents for E-commerce applications. In Section 3, an overview of multi-agent based E-health applications is provided. Section 4 covers existing literature on the use of multi-agents as applied to Intrusion Detection Systems. Section 5 discusses the use of multi-agents for securing Telematics and Transport Systems. Section 6 covers the use of multi-agents for Environmental Monitoring applications. In Section 7, a summary of all findings through the survey is provided. Finally, Section 8 highlights directions for future work in improving multi-agent systems.

2. Multi-agents for E-commerce

With rapid increases in the bandwidth of infrastructure networks during recent years, a surge in the number of commercial applications using them has been witnessed. The Internet has emerged as a major platform in supporting electronification of commercial activity, *aka* E-commerce. Examples of such activity include: online bidding, online buying and selling of items, and online banking. With growing consumer bases for these commercial services, it has become critical to assure security of public transactions for the customers. This is due to use of sensitive online information needed for commercial activity. Such information includes customer's social security numbers, mailing addresses, email addresses and above all, credit card details. In addition, there also exists a need to protect electronic cash and electronic cheques from malicious attacks by the adversary class, and to enforce information disclosure (Jin et al., 2010; Venkataram et al., 2007). This is needed to avoid any compromise of critical resources that could eventually lead to undesirable financial losses. It is therefore crucial to establish trust between the services and the consumers to facilitate secure transactions pertaining to an E-commerce application. An E-payment system is required to satisfy the key security principles of authorization, integrity, confidentiality and anonymity (Asokan et al., 1997).

A multi-agent system has been proposed in Guan and Hua (2003), by organizing agent competencies and functionalities in a hierarchical manner, to address the issue of protecting E-commerce infrastructures against security concerns and system failures. In the proposed architecture, several agencies are introduced to interact with each other under the jurisdiction of a centralized master agent called the *Agent Butler*, and provide services such as information collection, negotiation, decision-making, payment transaction and database maintenance. Within each agency, several layers of functionality is implemented to achieve the desired objectives. Three layers are defined as follows:

- *Service layer*: In this layer, the financing agency (responsible for E-payment) interacts with the other agencies, as well as with the Agent Butler, for coordination and information sharing.
- *Interaction layer*: This layer consists of the Payment manager, Account manager and the Auditing manager. The Account Manager is responsible for registration functions such as those associated with registration of new credit cards. Similarly, accountability of all transactions is maintained by the Auditing Manager, and the monetary transactions are executed through the Payment Manager.
- *Payment mechanism layer*: This layer consists of agents to perform actual payment transactions using one of several protocols such as Secure Electronic Transaction (SET) or Electronic Currency Management.

Lack of trust may lead to hesitance in the use of E-commerce systems (McKnight et al., 2002), thus leading to a reduction in the customer base. Much research has been done in examining the consumer perceptions of risk and trust, and the impact of malicious attacks on the consumer base of E-commerce applications. A scalable trust model for E-commerce systems is proposed in Zhao et al. (2008), wherein a mediating agent (transactional) performs a transaction between a buyer and a seller to facilitate a successful and a secured E-commerce activity. Given a set of n transactions agents, the key question is which one to trust more in carrying out a transaction between a buyer and a seller. The resulting system is capable of providing some degree of assurance to the customer on the security of online exchange of sensitive information, appertaining to the customer.

When multiple agents are present in a system, it is important to form a coalition (Ye and Yi, 2002) of a subset of the available agents (to be selected from tens of thousands), so as to be of any benefit to the entire E-commerce system. The authors also propose a scheme to ascertain confidentiality and integrity of all the messages exchanged between the coalition agents. A coalition signature scheme has been established using the concept of a public key infrastructure, wherein each agent possesses a pair of public and private keys. Using the private key, an agent generates signatures on the messages, whereas the corresponding public key (known to all other agents) is used by the receiver agents to verify the signatures, where, a signature is nothing more than a condensed form of the message (aka message digest), encrypted using the private key of the sender agent. The private key of an agent is generated by a trusted third party, i.e. the Certificate Authority (CA), defined as an entity responsible for the generation, maintenance, and verification of public key certificates for use on the Internet. During system initialization, the public key of an agent is associated with the agent's identity. The CA is only activated when a new coalition is formed or if an old coalition is dissolved. The identity of an agent a_i, given by ID_i, is derived from either the name of the agent or from the index of the agent. Through encryption of all data communicated between each agent pair, the core security property of *confidentiality* is ascertained between communicating agents belonging to the coalition. It is rather very time consuming to encrypt a message using

public–private key pairs (Wang et al., 2008). Therefore, a symmetric key encryption is also proposed for establishing a common secret key between a sender and a receiver using the sender's or receiver's public–private key pair. For example, a sender can send a symmetric key using a receiver's public key, which can only be decrypted by the receiver using its private key. Once this is done, then the symmetric key is subsequently used to encrypt the message, which can only be decrypted by the agent holding the corresponding private key. Thus, the *integrity* of individual messages that are exchanged between the coalition agents can be sustained by means of computation and verification of signatures of the message (message digests encrypted with a symmetric key), for protection against message tampering, by having the hash value recomputed at the receiving end, for comparison with the received signature.

In Song and Korba (2002), the authors identify the need for efficient management of multiple agents to ensure scalability of agents in the presence of thousands of agents, in E-commerce systems. Through simulation experiments it is observed that the use of public key cryptography in a multi-agent system has a severe impact on the scalability of the system. When public key encrypted messages are exchanged between the agents for purposes of confidentiality or authentication of messages, a linear increase in the total processing delay is observed, as compared to a negligible delay observed for the non-cryptographic approach. Experiments were also performed to test the effect of using the Secured Internet Protocol (i.e. IPSEC) (Kent and Atkinson, 1998) on the scalability of the scheme. It was noted that the use of symmetric key cryptography (as part of IPSEC) has little effect on the system scalability if used for the confidentiality and integrity protection on the exchanged messages between the agents. A near linear total processing delay is observed for both the IPSEC as well as the non-IPSEC inter-agent message exchange. It can thus be conclusively stated that the common notion of decreasing efficiency for increasing resilience of a multi-agent system to malicious threats in E-commerce systems is strengthened with the findings appertaining to the use of public–private key pairs and symmetric keys, as provided in Song and Korba (2002).

3. Multi-agents for E-health applications

In general, E-health is defined as the use of technology, and the Internet in particular, to provide efficient health care services (Bergenti and Poggi, 2009). As a consequence, not only is the cost of providing such services reduced, but invariably the effectiveness of service is improved manifold. Considering the significance of E-health applications, recent research has intensified to improve the performance of such systems. Health care systems generally consist of discrete and heterogeneous entities, with a significant demand for remote access by both health care providers as well as by consumers including medical practitioners and patients. Multi-agent systems are primarily distributed in nature, and the presence of coordination between heterogeneous infrastructures of E-health systems is practicable through the use of such systems. The data handled by a health care system includes patient's medical history, outcomes of diagnosis, test results and personal details. The sensitive nature of the data entails the need for having a secured mechanism for protecting unlawful disclosure or modification of data stored and communicated in E-health systems.

To protect confidential E-health data from unauthorized access, a role-based access control scheme is proposed in Li and Hoang (2009). The scheme is dynamic in terms of interactivity between three key elements, namely, *role*, *interaction* and *organization*. The E-health system is perceived as a multi-agent system, with different personnel playing different roles under different organizational contexts, and varying interactions being required amongst themselves. The role is defined as a peer-to-peer model, capable of receiving requests from other entities as well as initiating requests for other roles in the system. A role that initiates a request to another role is defined as an *initiator*, and if a role receives a request, it is called a *reactor*. Each role associated with a security property is also defined as a dependency, which identifies the constraints spelled out to be met by a dependency. These constraints represent a set of conditions that restrict or allow entities belonging to a particular role, permission for doing a task, such as data access. An illustration of such a model is given in Fig. 1. The dependencies are classified into four classes given by:

- *Open security dependency*: The interactions between the initiator and the reactor are examined to meet the security dependencies, but tolerable otherwise. Even if certain aspects of the dependency are not met, the overall security of the scheme is not compromised, and therefore, such missing conditions can be tolerated.
- *Initiator security dependency*: The initiator presents a minimum security requirement to the reactor that must be met if these requirements are to be classified as valid.
- *Reactor security dependency*: The reactor presents a minimum security requirement to the initiator, and the initiator must meet these requirements to be classified as valid.
- *Bidirectional security dependency*: Both the initiator and the reactor present their security constraints to the other entity, and both must meet the requirements, spelled out at system initialization time.

The interaction model is defined as a template that distinguishes the type of access to be given to a certain entity for a particular task. This model is subdivided into two categories. The *closed* category is defined as having a fixed number of participants for a particular interaction that cannot be changed for that particular interaction. For instance, if an initiator requests a reactor for retrieval of a patient record, no intrusion by a third party will be allowed into this interaction. An *open interaction* is where the number of participants in the interaction can change with the progress of the interaction.

The proposed organizational model specifies the structure of a given health care organization, the organization rules and organizational positions. The designations of entities or personnel within an organization together with the set of rules governing the type of access to be given, is required for the E-health system. For instance, a person may act as either a patient, a general practitioner or a specialist. For each particular role, a set of policies are defined to exclude or include the entity from access

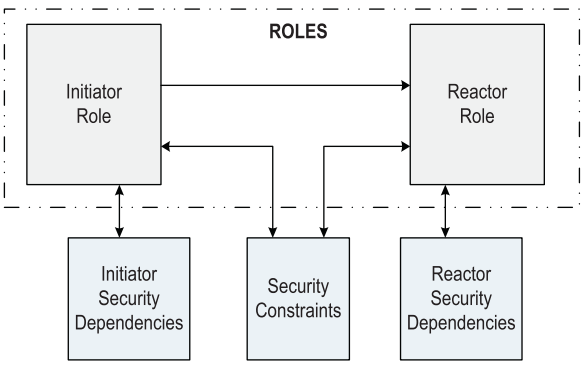

Fig. 1. A role-based access control system for E-health.

to a given data object. The organizational model is therefore categorized into three categories, namely, the position of the subject, rules, and organizational structure. The position of a particular subject (such as patient) defines the rules that will be imposed on him or her. These rules dictate the policies and constraints on how information flows in, out or within an organization, to the subject in question. The structure of the organization defines the topology, given by either centralized, peer-to-peer, multilevel hierarchy, or complex composite structure. Depending on the topology of the organization, the roles of the subject will adapt to the needs and constraints imposed by the defined rules. The interactivity and collaboration of the various subjects of the organization under the constraints of the rule set define the security of the multi-agent E-health system.

A secure platform is introduced in (Ulieru and Ionescu, 2006) for provisioning E-health services by having secure web data manipulation by medical professions on the patient records. The overall objective is to facilitate access to a large volume of distributed databases, which deal with patients having diseases that require specialized knowledge, and are retrievable from the databases. As a result, the integrity of all sensitive patient data is guaranteed, provided that the health care practitioner ensures strong password protection practices (such as the use of unguessable passwords and regular password changes), for access and manipulation of patient data.

4. Multi-agents for Network Intrusion Detection Systems

Computer networks are prone to a plethora of malicious attacks. Such attacks may be launched by the adversarial class by means of exploiting vulnerabilities of network protocols, with the intent of preventing access to the underlying network as well as to the services provided therein. As a consequence, a valid service to an end-user is disrupted. Examples of some attacks include IP spoofing, TCP SYN, ICMP Echo attacks, Doorknobs and Traffic Analysis Attacks (Stallings and Brown, 2007). Generally, the first line of defense against a malicious attack is through attack detection. For this purpose, an increase in the deployment of *Intrusion Detection Systems* (IDS) has been observed for attack detection purposes. An IDS may be either a software-based solution or a piece of dedicated hardware, installed on a computer network, for continuous monitoring of traffic flow, and subsequent detection of malicious activity. An IDS can be classified into two common types (Anderson, 2001):

- *Host-based*: These are designed to monitor mal-activity on a single host on a network, by analyzing the local computer logs, system calls, and application-activity.
- *Network-based*: This type of an IDS analyzes the flow of network traffic to detect malicious activity. Generally, this type of IDS is installed at an entry point to the network such as on a hub or a switch, so as to observe all traffic flow in and out of the network.

Some additional classifications such as protocol-based IDS, application-based IDS and Distributed Host-based IDS may also be found in the literature.

Considering the multiple entry points to a network, the need for having distributed and multiple IDS' operating in the network cannot be understated. At the same time, it is essential for all operating IDS' in a computer network to cooperate and share their respective findings, so as to construe a holistic view of the activity, malicious or otherwise, as illustrated in Fig. 2. In Boudaoud et al. (2000), the authors propose an intelligent multi-agent based approach for the design of an IDS. The proposed scheme operates at two layers, namely, *Manager* and *Local*. The Manager layer operates at a higher level and manages the overall security of the network. Three types of agents operate at this layer, namely, *Security Policy Manager*, *Extranet Manager* and *Intranet Manager*. The Security Policy Manager agent is responsible for managing the policies specified by the network security administrator. The Extranet Manager agent only manages Intranet Manager agents, and is responsible for assigning and delegating intrusion detection tasks to the Intranet Managers. The operations of the Extranet Manager agents are controlled by the policies of the Security Policy Manager agent. The Intranet Manager agents control local agents that monitor the network traffic flow and report to them. The *Local* layer contains a set of Local agents, with each agent being assigned a specific function. The Manager layer specifies the list of tasks that need to be accomplished by the local agents of the local layer. In return, the local layer sends its responses back to the Manager layer, based on the observation of the network traffic, for further action. Each agent is constituted of seven different modules that help the agent act effectively towards task accomplishment. The modules are:

- *Perception module*: The information that can detect an intrusion attempt is gathered by this module, by means of network traffic tapping and filtering.

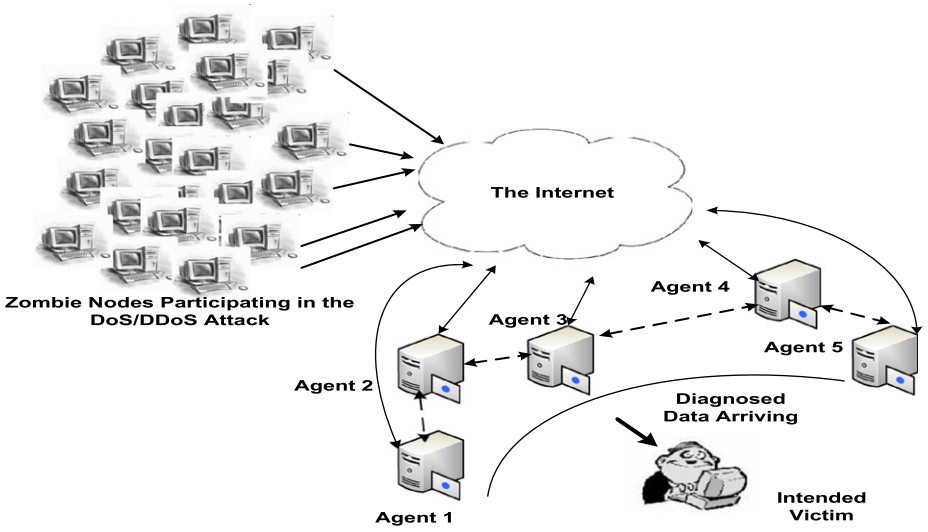

Fig. 2. Multi-agent based coordinated intrusion detection system for computer networks.

- *Communication module*: Allows multiple agents to share their findings and decisions.
- *Action module*: Upon detection of an intruder, this module performs an action such as deactivation of a particular host.
- *Report generation*: Establishes reports on the detected malicious activity, and sends them to the network administrator.
- *Deliberation module*: Facilitates the use of intelligence in decision making and deliberation activity by each agent. This module may rely on the use of existing knowledge of known malicious activity, and may also enhance its intelligence by active learning through techniques such as Neural Networks.
- *Interface module*: Interacts with the security administrator to receive policy-related specifications and to deliver reports on the network activity.
- *Supervision module*: Coordinates the interaction between the above-mentioned modules using finite state automata.

An attack pattern ontology and a formal framework within a distributed multi-agent IDS is discussed in Kolaczek and Juszczyszyn (2007). It consists of two types of agents, namely, *Monitoring* and *Managing*, for a set of nodes, such as host machines in the network. The monitoring agents are responsible for observing the nodes for both ingress and egress network traffic flow. The agents further process the captured traffic and draw outcomes of their respective analysis to jointly agree on the state of the network with regards to intrusion activity. The managing agents gather information from the monitoring agents to help formulate a holistic view of malicious network activity. The resulting outcome is the generation of reports on the attacks. The proposed design, as illustrated in Fig. 3, relies on the following parameters collected by the monitoring agents:

- Source and Destination IP addresses,
- Source and Destination Port numbers,
- Number of bytes sent and received, and
- Ratio of the number of Synchronize (SYN) packets to Finish (FIN) packets.

To measure any changes in the observed IP addresses and their corresponding port numbers, the proposed scheme computes and analyzes Shannon's entropy values. In particular, the following variables are extracted from observed network traffic for further analysis: Source IP address, Destination IP address, Source Port number, and Destination Port number.

The value of the entropy is calculated based on the standard Entropy calculation equation given by

$$e = -\sum_{i=0}^{N}(p_i \log p_i), \tag{1}$$

and,

$$p_i = \frac{n_i}{\sum_{i=0}^{N} n_i}, 0 \le i \le N, \tag{2}$$

where N is the number of distinct IP addresses/port numbers, n_i—number of packets with a particular source/destination IP address/port number observed in a specific time window, and $\sum_{i=0}^{N} n_i$—total number of packets observed in a time period t_i.

Entropy values change only when there is a change in the pattern of communication packets observed. As a result, any malicious activity with abnormal pattern of traffic flow will yield high Entropy values within a given frame of time, thus alluding towards a possible intrusive activity in the network.

A distributed multi-agent system is proposed in Orfila et al. (2005), for intrusion detection. The architecture consists of three types of agents: *predictor*, *evaluator* and *manager*. The hierarchy of the decision-making process in the proposed architecture is illustrated in Fig. 4.

The predictor agent is to predict whether an intrusion attempt took place or not, and operates based on the instructions received from the evaluator agent. Based on their past decisions, the

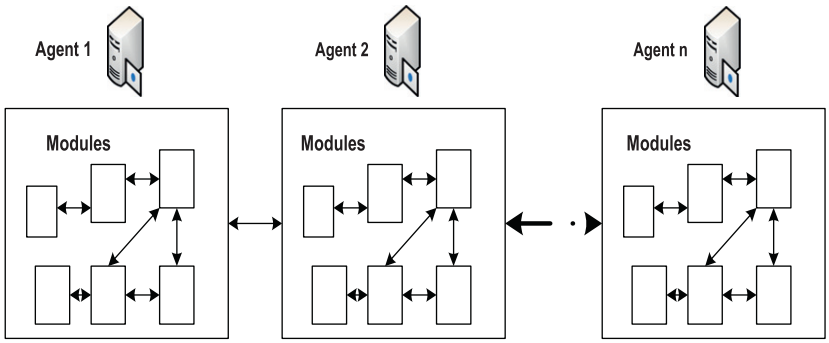

Fig. 3. Collaborating multi-agents for Network Intrusion Detection.

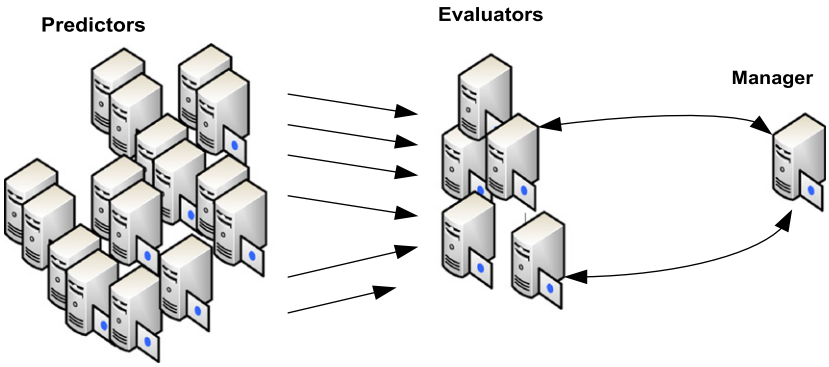

Fig. 4. Three-agent architecture for Network Intrusion Detection.

evaluator agents are responsible for weighing the level of trust of each of the predictor agents. If a predictor agent has made more correct guesses in the past as compared to another predictor agent, then it will be assigned a higher weight by the evaluator agent. The final decision as to whether an intrusion attempt was detected or not rests with the manager agent. The manager agent runs in a dynamic environment, constantly learning and updating the system parameters. The manager agents are analogous to post-mortem agents, with the task of performing an aggregated decision derivation, based on observed readings from the evaluator and predictor agents. The proposed scheme is intelligent and adaptive, as the weights assigned by the evaluator agents to the predictor agents are dynamic and vary based on the performance of the predictor agents. Given that H stands for the probability of detecting an intrusion whilst an intrusion attempt actually took place, and F stands for the probability of declaring an intrusion in the absence of any intrusion attempts, the weights assigned by the evaluator agents are derived through the following equation:

$$weight_i = \frac{H_i - F_i + 1}{\sum_j (H_j - F_j + 1)}. \quad (3)$$

As a result, the reputation of a particular predictor agent increases if its hit rate is higher, and the false alarm rate is lower.

A reinforced learning-based approach towards the design of an intelligent multi-agent IDS is proposed in Servin and Kudenko (2007). As part of the reinforced learning architecture, each agent shares information by sending signals to other agents. The agents sense their environments in discrete time steps and map the inputs to a local state information. The agents participating in such activity are termed as reinforced learning (RL) agents. These RL agents execute actions and await feedback from a higher authority, or a manager agent. After receiving a reward for a particular action, the agents observe any changes in the environment, and update their policies to optimize the reward received for future actions. The proposed architecture contains distributed sensors that process local state information and pass on short signals up the hierarchy of RL agents. Through these signals, the RL agents learn to distinguish between normal and anomalous activity. The RL agent decides on a signal action by collecting signals from all the sensor agents. If a signal generated by the RL agent is in accordance with the actual state of the environment, either normal or anomalous, then all sensor agents receive a reward from the RL agent. However, if the RL agent generates an incorrect signal, then all the sensor agents receive a negative reward.

Each agent uses the Q-learning technique which is executed in a particular state. Through this method, the agent recursively tries to estimate the value function. A Q table is thus formed with rows as states and columns as actions. The agent in each state s chooses an action a, and observes the reward r and the next state s'. It subsequently updates the Q value, given by $\hat{Q}$. The resulting equation is given by

$$\hat{Q}(s,a) \leftarrow \hat{Q}(s,a) + \alpha(r - \hat{Q}(s,a)). \quad (4)$$

where $\hat{Q}(s,a)$ is an update to the existing $Q(s,a)$ value, α is the normalization factor, and r is the reward for selecting an action a.

The authors considered a simple experiment, wherein two sensor agents and one RL agent operate in one of two states. The sensor agents need to learn to produce a correct signal action sent to the RL agent, and the RL agent needs to learn to interpret the received signals, before an update to the state takes place.

5. Multi-agents for Protecting Telematic and Transport Systems

Telematics is the science of using a telecommunication network or device to send, receive and store information. The study of Automotive telematics is specific to the use of telecommunication networks and devices to which vehicular based commerce services can be deployed. Examples of applications include vehicle tracking, emergency roadside assistance, tracking lost or stolen vehicles, and pay-for-use rental and insurance services. In order to provide these services, data parameters pertaining to a vehicle, such as the location of the vehicle, need to be collected regularly. Since the information regarding the location of a vehicle or the details of the driver may be confidential in nature, there exists a need to protect the privacy of all such data, so that secure access to telematic services can be assured. If such information is leaked, it may lead to unfair competition of the vehicle's owner with rivals and other competitors. In addition, there also exists a likelihood of tampering of all such data. Thus, it is essential to protect both the confidentiality as well as the integrity of all data associated with telematics, including vehicular telematics (Stoian et al., 2006).

A multi-agent based framework is proposed in Duri et al. (2004), to provide both confidentiality as well as integrity for automotive telematic services, without affecting the efficiency of the service infrastructure significantly. The proposed architecture addresses the issue of data confidentiality and integrity by allowing parties interested in acquisition of locomotive information to form a trusted system with confidential data, based on deployed agents. As a result, the interested party is allowed to use the private data without having the data leave the trusted system. The trusted system is essentially a tamper-resistant piece of hardware situated on a vehicular object, to protect data from any physical or logical modifications by an adversary. The vehicles are equipped with wireless devices, sensors and a computer. The sensors are needed to collect vehicle data such as the temperature of the engine, performance metrics and safety information. In addition, Global Positioning Sensors (GPS) are also deployed for allowing remote acquisition of vehicle location coordinates.

Agents are deployed by the service providers or third party vendors directly into the vehicle's computing system, and perform tasks, to which the driver has subscribed to. For instance, if the driver has subscribed to an insurance plan, he or she will have an agent deployed into the vehicle computing system, which will calculate vehicular parameters such as mileage and report back to the insurance company. In addition, another agent may be deployed for roadside assistance that facilitates tracking of the vehicle coordinates in case of a breakdown or an emergency. In the proposed framework, the data acquired from the agents is not readily available to all requestors due to the privacy agreement between the vehicle's owner and the service providers. Essentially, all requests are checked for privacy compliance, and only upon verification, is the vehicular data released to requesting entities. An illustration of the proposed framework for this application is given in Fig. 5. The mileage calculator computes an estimate on the number of miles traveled by a vehicle on various highways, based on information that it receives from the GPS sensors on the vehicle. The Diagnostic Aggregator subscribes to the vehicle data, and is thus updated with changed information on a regular basis. The Privacy Manager is responsible for verifying the authorization of a requestor to access the requested data. The policies devised initially are compared against by the Privacy manager for compliance purposes. The Repository Manager is subsequently communicated with to provide 'grant' or 'deny' access to the requestor of data, based on the input of the Privacy Manager. The Repository Manager subsequently composes a response message to the requestor's query. The Message Broker is responsible to send out notifications to the data subscribers, informing them of the outcome of their respective requests.

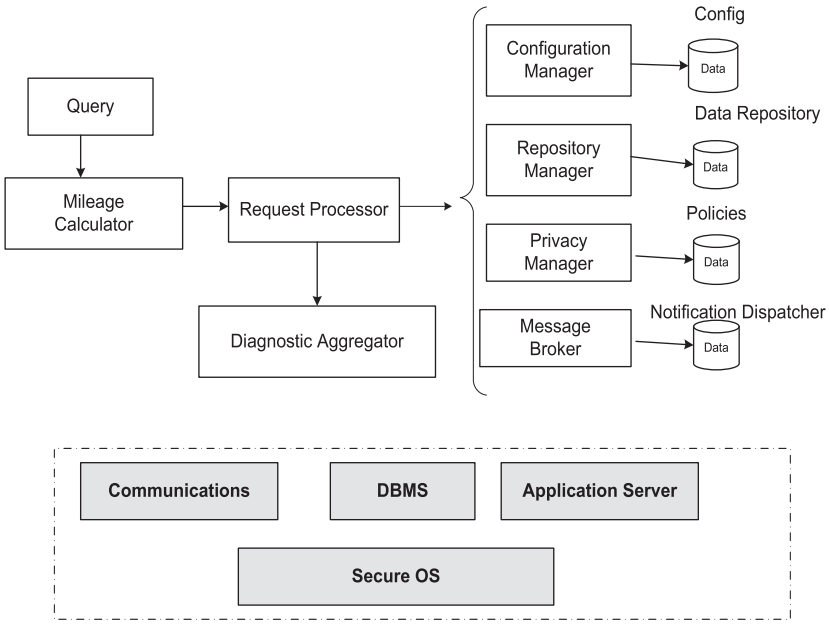

Fig. 5. Secure telematic system design through collaboration between multiple agents.

A secure multi-agent based approach is used in Fricke et al. (2001), to provide telematic services, for efficient use of telecommunication networks. The security features of the Java Intelligent Agent Componentware (JIAC) are defined in the proposed architecture. At the agent-level, each agent can use a protocol provided by JIAC, similar to Secure Sockets Layer (SSL), to encrypt or digitally sign communication signals at the application layer, prior to being communicated between the agents. Using JIAC's service control mechanisms, access control lists can be created by attaching service control lists to services. Therefore, the receiver of a secure (encrypted, signed) message can decide whether the originator agent has access rights to a particular service or not. Additionally, three different types of security-specific agents are also available in the proposed design: a certificate authority (CA) agent, a security agent and a security service agent. The CA is deployed by a user to issue certificates based upon human requests received. It always resides on its secure agent location. The CA is responsible for forwarding certificate-related information to a secure agent. The secure agent is a domain of secure places for all agents. It can be requested by another agent such as CA agent for provisioning of a secure agent place. Secondly, the secure agent is responsible for defining and maintaining the security relationships between the different secure agent places. For instance, an agent pair $\{a_1, a_2\}$ may have a one-way 'a_1 send messages to a_2' relationship. In such a scenario, a single public key of a_2 to encrypt all a_1 messages, will suffice for securing the communication between a_1 and a_2. The security service agent is responsible for creating time stamps, signature verification, and to act as a trusted third party in a contract agreement as part of an E-commerce application. As a result of a hierarchical approach towards securing the proposed design, an efficient agent-based approach is thus proposed for secure inter-agent communications.

Optimal traffic signal control for improving urban traffic and reducing driver delays is an area of concern in all metropolitan cities during present times. Through the architectures proposed in Balaji and Srinivasan (2010), an effort has been made towards providing effective signal controller operations, to protect a safe urban traffic flow mechanism in a large metropolis, namely, Singapore. The scheme operates by having each traffic intersection controlled by an intelligent agent, which decides the desired policy for its signal timing, based on congestion information that it receives from other operating agents. The data is collected by loop detectors which are placed at pertinent locations on the streets, and provide an insight on the traffic data, including, queue length, traffic flow, occupancy and the total number of vehicles present on a traffic path at any given time. External factors such as environmental conditions, driver behavior, and vehicle length, add noise to the acquired data. The proposed signal control architecture consists of three layers, each one being distinct in its operation. The lowest layer consists of intersection controller agents (ICAs) for controlling the traffic intersections. The middle layer consists of several zone controller agents (ZCAs), to control a cluster of ICAs. Finally, the top layer consists of a regional controller agent (RCA), to control all ZCAs. An illustration of the proposed multi-agent traffic control system is given in Fig. 6. The daunting task of traffic management on a large scale is divided into several subtasks, with each agent deemed responsible for a particular task or a group of tasks. It may be noted that all agents have access to environmental data, and are capable of autonomous decision-making. In each layer, the agent decides the appropriate local policies to control green signal timings, during a given period of time. In addition, agents interact with each other based on a pre-defined model, so as to construe a larger picture of the traffic, and to make better decisions. A policy repository stores all policies for inter-agent communication in the traffic network, at the end of each time phase. Older policies are phased out and newer policies are updated, and conflict resolution is also performed, all within the policy repository. An arbitration process is executed to ensure that higher layers are given precedence in policy implementation. However, since all lower layer agents affect higher layers, the hierarchy provides an even balance between policy definition and its implementation thereof.

The ZCAs operate by performing neuro-fuzzy operations on all collected data, to decide on the time duration of a green traffic signal, for a specific phase of time. The ZCA takes as input an intersection's occupancy, traffic flow rate, rate of change of traffic flow, and a cooperation factor generated by the ICA, and provided as input to the ZCA. All these four fuzzy sets have three linguistic labels: *high*, *medium*, and *low*. A set of fuzzy rules are defined. An example fuzzy rule is as follows: *IF intersection A has HIGH traffic flow and intersections B and C have LOW traffic flow during Time*

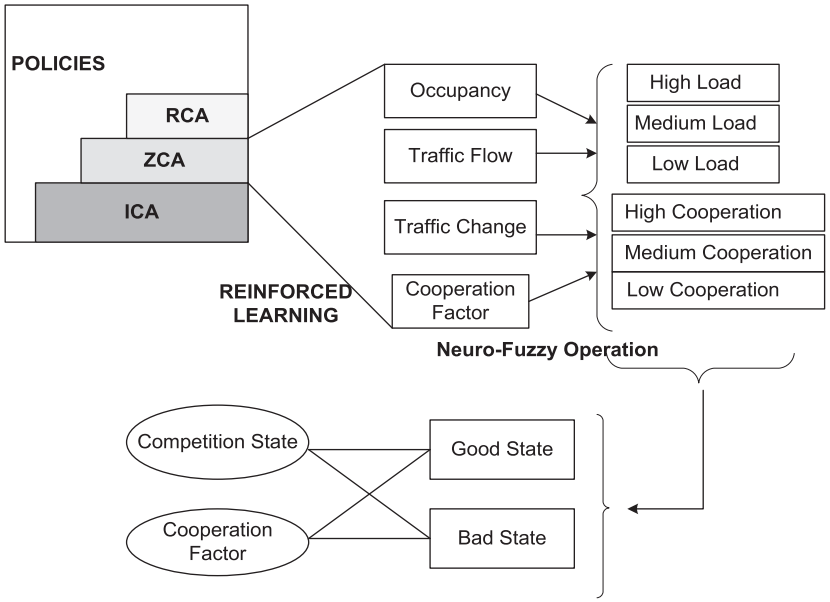

Fig. 6. Protecting urban traffic controls through multi-agents.

Phase τ THEN switch Signal A to Green for n Seconds. The third layer of the ZCA fires each fuzzy rule based on the T-norm fuzzy operation, with the *minimum* operator. Nodes in the third layer define the degree of traffic load in a given zone under the jurisdiction of a given ZCA. The fourth layer nodes represent outcomes corresponding to fuzzy rules of the decision module. The outcomes include: the level of cooperation necessary between various agents of the hierarchy, and various signal improvement and control policies that may be executed during a given phase of time.

As a result, the proposed scheme provides a distributed multi-agent based approach towards safe and smooth urban traffic flow in large metropolitan cities.

6. Multi-agents for Environmental Monitoring

The effort towards reaching a state of electronic device omnipresence has bore fruits in recent times with the rapid advances in processor and wireless communication technology. At the same time, with processor technology approaching sub-micron levels, wireless sensor nodes have evolved to become a potential force towards facilitating remote monitoring and wireless communication of readings to central base stations. Applications of such a technology are critical in nature. Some of these are: bushfire monitoring, temperature monitoring for weather forecasting, monitoring motion of adversarial vehicles in a battlefield, and motion sensing for surveillance as well as to switch on/off electrical devices to conserve electricity.

Multi-agent systems provide the necessary platform for such sensing devices to remotely operate, observing their respective environments, and conveying their respective readings either to the centralized base station, or for sharing with other peer nodes. Sensor nodes generally operate in unsafe environments (Culler et al., 2004), and monitor critical data relevant to applications with implications that may affect hundreds to thousands of individuals. For instance, a bushfire alarm, if generated in a timely manner based on sensory readings, can save the lives of all inhabitants of a region in question. In Abielmona et al. (2006), a tree-in-motion mapping (TIMM) technique is proposed, to intelligently combine individual sensor data, while reducing the entropy (or randomness) that may be observed therein. As a result, a multi-resolution and multi-dimensional view of the environment being monitored by the sensor nodes is presented. Since the data is acquired from multiple sources, such an approach is also referred to as *multi-sensor fusion*. Two classes of such an approach are: *distributed* (D-MSF) and *centralized* (C-MSF). The former provides a platform for sensors to transmit locally observed readings amongst each other, process information locally, and synthesize a global estimate of readings from individual estimates. On the contrary, the latter approach (C-MSF), requires all sensors to transmit local readings to a centralized base station (or another central processor), which performs information fusion, as well as global estimation. Each agent, regardless of it being operational in a centralized base station or on an individual sensor, is programmed with a local model of expected readings from its environment. As a result, any deviation, also defined by the authors as a measure of Entropy, is observed and reported accordingly. The entropy associated with each cell of the environment, is defined using Shannon's information entropy measurement model, in terms of bits, for a discrete random event X, with possible states $1, \ldots, n$, as illustrated in Eq. (1).

The purpose of fusing sensory readings is to ascertain that the randomness that may be observed in readings of a given sensory region does not occur due to a malfunctioning sensor node, but rather is a phenomenon occurring due to the collective observation of a set of sensor nodes operational in the environmental cell. The proposed technique is tailored for resource-constrained devices, as it incurs very low computational overhead, and storage needs on individual sensor nodes.

In Abielmona et al. (2011), a robotic sensor agent-based architecture is proposed for monitoring territorial security. The multi-agent system architecture proposed is intended for protecting international borders, transportation, as well as other critical infrastructures. The proposed architecture is termed as a *Retro-active* architecture. An event occurring in the environment causes a momentum in behavior to increase, which leads to the behavior reacting to the event by executing a behavior *plan*. An event essentially is the occurrence of a phenomenon in the environment that requires attention. Through the process of *momentum resolution*, it is decided by the agent as to which behavior to invoke based on the event occurrence, and how the behavior executes, through *plan execution*. A feedback mechanism ascertains that the

appropriate reaction to an event is invoked, and such an environmental observation is recorded (i.e. learned) for future assertions. As a result, the architecture allows an autonomous robot to learn over a period of time, based on reactions to event occurrences, ultimately preparing these robots to respond in real-time to actual events in the environment, as they occur.

In order to protect a critical resource such as an airport, railway yard, public transport system, or even international borders, from unwanted trespassing, territorial security is an essential component of environmental security that cannot be overlooked. In Technologies and Platforms, the authors present challenges faced when wireless sensor networks and fusion of generated sensory data, is applied for ensuring territorial security. For such an application, sensor nodes perform sensing of their respective local environmental cells, to collect data. For instance, motion sensors may be programmed to collect any activity at the perimeter of a critical facility, or an infrared camera can detect trespassing into a restricted facility. Moreover, sensor nodes need not be stationary, as they may be placed on a mobile object such as an armed vehicle in a battlefield. The lack of a robust communication media to accommodate bandwidth-savvy sensory data transmissions between the sensors and a centralized server demand the presence of localized data preprocessing and filtering on the sensors themselves. The authors propose the Larus Intelligent System for fusing data streams in real-time, to issue alerts based on anomalous findings in the data. As a result, a decision support system emerges, through collective data generation by the sensor nodes, and subsequent data fusion as well as mining. Moreover, the proposed scheme is dependant on existing data visualization technologies based on the Extensible Markup Language (XML) and Semantic Web, with options for future extensibility. The data fusion techniques that may be programmed on individual sensor nodes can be changed based on preferences and requirements of the application in question.

Through intelligent cooperation of multiple sensor nodes, the task of monitoring a distributed, and geographically dispersed area, has become a reality. If each sensor node is considered to be a single agent, programmed to perform specific tasks, associated with filtering and fusion of observed readings, the proposed scheme can prove to be an effective solution for providing large-scale territorial security.

7. Summary

Contemporary critical infrastructures are prone to failure due to their large coverage, presence in harsh environments, as well as due to system vulnerabilities that may be exploited by the adversary class for malicious activity. Therefore the role of effective and efficient techniques based on the concept of collaborating multi-agents, to protect such systems cannot be understated. Recent advances in the fields of distributed, autonomous and intelligent computing have paved the way for the use of the multi-agent paradigm for protecting critical applications from both failure as well as security threats. The analysis presented provides an insight on the use of distributed information processing techniques and multi-agents for efficient data communication and processing, without compromising the level of protection that such multi-agents can provide to the aforementioned critical infrastructures.

In general terms, to protect critical infrastructures, a multi-agent based system design will depend on two key factors:

1. The ability of the multi-agent system to efficiently monitor, communicate, and generate decisions, based on readings from individual agents.
2. Assurance of confidentiality and integrity of all data stored and communicated between collaborating multi-agents.

The disparate nature of tasks performed by individual agents of a multi-agent system, require local processing of all data, so as to reduce the overhead associated with large scale data transfers that may be required as part of inter-agent collaboration, especially for those involving light-weighted devices. Such communication is essential to reconstruct a holistic visualization of the state of the application running on the multi-agent system. For instance, a record-monitoring agent in an E-health application may require communication between the application modification agent active on the desktop of a health care practitioner. If data can be pre-processed (or filtered) on a need basis prior to it being communicated to the application modification agent, a significant reduction in the bandwidth utilized by the multi-agent system, is achievable. More so, if the frequency of inter-agent communication is high. Similarly, remote readings collected by a set of sensors, operational on a vehicular telematic system, would require wireless communication of all collected data over long distances, with intermittent transmission capabilities i.e. incurring high error rates of transmission. With high bandwidth of contemporary computer networks, at 100 Gbps, the rate of arrival of data packets is in the order of 2 million per second. Therefore, the need for rapid information processing for detecting anomalous network traffic by a distributed intrusion detection system is a necessity.

Multi-agent systems operate on the principles of distributed information processing and collaborative decision making. The agents that comprise a multi-agent system for protecting a critical infrastructure may either be operational within a single computing node, or may be geographically distributed across several computing nodes, carrying out their designated tasks based on the application needs. Therefore, the need for constant collaboration for information exchange between the agents to constitute a holistic visualization of the state of a system, in terms of individually observed readings, is essential.

E-commerce servers operate in distributed fashion, by having mediatory agents operational between the clients and the servers, for secure transaction execution. Certification authorities are utilized for establishing, verifying, and maintaining public–private key pairs, essential to ensure confidentiality and integrity of exchanged messages between the distributed agents of such a system.

E-health systems have scattered agents associated with individual components of the system. Some agents may be associated with the execution of tasks on the computing node of the medical practitioner, whereas others might be associated with the secure transfer of data to and from between the medical practitioner computing nodes and back-end patient databases. Secure and remote access to all information generated and communicated between these diverse and distributed set of agents in the E-health system is thus attained through the use of either public–private key pairs, or through symmetric key encryption.

Network-based Intrusion Detection Systems operate through the collaboration of multiple agents placed at various egress points of a computer network, so as to ascertain that no data packet sneaks into the network without being monitored. Distributed diagnosis of egress network traffic packets, and subsequent cooperation between the agents, to construe a holistic visualization of the state of the network, is essential for timely and accurate detection of malicious network attacks.

Multiple agents are essential for remote and geographically distributed monitoring of telematic and transport systems, through the sharing of sensory readings depicting either the geographical location of a logistic entity, or the state of traffic in

a metropolitan city. Collaborative information processing from multiple arms of the telematic or transport network is therefore essential to provide for an accurate analysis of the state of the system, or for decision-making in terms of traffic diversion, essential for transport system management.

The success in Environmental Monitoring is only accomplishable through distributed monitoring of the state of the environment in terms of parameters to be assessed (such as temperature, pressure, humidity, etc.). Distributed multi-agents sustain such need through distributed monitoring through sensors and actuators, and subsequent collaboration, to provide for a global picture on the outlook of the environment based on the observed parameters.

In order to ensure adequate security of critical applications such as those belonging to the five categories detailed in this paper, a common practice for securing multi-agent systems from unwarranted and unlawful access and modification is to use data encryption. Public-key certificates, the public key infrastructure (PKI), and all accompanying requirements for such a system to be deployed, is one common approach emphasized and actually enforced in most if not all schemes surveyed. The security provided through such an approach is fairly robust in terms of effectiveness, and universal scope of enforcement, albeit at the cost of significant resources needed for establishment of such an infrastructure, and its subsequent application-specific usage. The cost appertaining to establishment of distinct public keys for a typical PKI is in the order of $O(2nk)$, where n is the total number of agents in the system, and k is the length of the key pair. The PKI operates by having a key pair (public, private) generated for every agent of the multi-agent system, required for secure data transfer over untrusted communication channels. The public key is globally known to all other agents, whereas the private key of an agent is not disclosed to any other agent. Any agent wishing to communicate with another agent will encrypt the message with the receiving agent's public key. The receiver agent is thus the only agent in the system capable of retrieving the original message from the encrypted message, using its private key.

8. Future directions

Although the PKI serves the requirement for provisioning security in a distributed multi-agent system, an alternative approach is the use of pairwise symmetric keys for establishment of secure communication channels, wherein the mutual agreement between all agent pairs can ensure that a shared symmetric key is exchanged at system initialization time. Such an approach would ascertain a significant reduction in the overhead associated with the establishment and maintenance of a public key infrastructure, CA deployment, etc. The existence of an integrity check mechanism in place to ascertain the origin and authenticity of individual messages exchanged in a multi-agent system is a relatively low overhead operation that may impose minimal processing power and lesser delays. However, the need for a secret key or a public–private key to ensure the use of signatures is unavoidable. The delays associated with integrity checks are relatively low as compared to PKI-based data encryption and decryption.

For rapid establishment and maintenance of keys between the agents, the use of key pools can be an effective solution. Through the use of key pools, a random set of keys is selected by each agent, and for securing communication between any agent pair, it is first ascertained that they share a common key. The Probability Pr[P] of having two or more agents share a common key is defined in Eschenauer and Gligor (2002). If a common key cannot be found for an agent pair, the agents look for other neighboring agents with whom they share a key. A communication chain can thus be established, for ensuring that inter-agent data communication is secure regardless of the number of agents that constitute a part of the chain. It may be noted that for such a scheme to successfully function, the underlying assumption is the presence of *equal trust* between all agent pairs of the entire system.

For purposes of data dimensionality reduction at the source of data acquisition, it is essential to have an in-network data preprocessing framework in place to provide localized data reduction capabilities to individual agents operating to protect critical infrastructure, operating under constraints of high accuracy and minimal delays. Several such data processing frameworks have been proposed in the literature to counter the issue of data dimensionality reduction. One such technique based on the principles of associative memory overlays is the Graph Neuron (Khan, 2002; Khan and Mihailescu, 2004; Baqer et al., 2005; Baig and Khan, 2008). The graph neuron overlay allows for individual agents belonging to the critical infrastructure application to store subpatterns to represent readings or observations of individual agents, that may eventually be merged with similar subpatterns of other nodes (or agents), to construe a holistic visualization of the state of a given application, or the environmental readings obtained. Each subpattern can be perceived as a threshold value of a particular system parameter (such as the level of activity on a border observed by a sensory agent), that needs to be referred to prior to any individual decision-making by a single agent. Subsequently, all such individual decisions or observations may be exchanged between the collaborating agents, to construe a holistic view of the system state. All learned subpatterns can be *recalled*, based on a simple comparison of stored threshold subpatterns of application behavior with readings observed by sensory agents from their environments. The entire agent data need not be communicated each time a requisition is received by another agent. Rather, a reduced version of the data can be communicated to the requesting agent. Although, the graph neuron suffers from false alarms, the experimental work found in the literature proves that the gain in performance through reduction in data dimensionality far outweighs the false alarm rate of the pattern matching approach towards data dimensionality reduction.

Therefore, the possible future step towards consolidation of distributed agent knowledge from large number of operational agents through localized information processing and secure and efficient collaboration can be felt as a perceivable reality.

Acknowledgement

The author wishes to acknowledge the continued support for research provided by King Fahd University of Petroleum and Minerals.

References

Abielmona R, Petriu E, Whalen T. Multi-agent system information fusion for environment monitoring. In: Proceedings of the instrumentation and measurement technology conference; 2006.

Abielmona R, Petriu E, Harb M, Wesolkowski S. Mission-driven robotic intelligent sensor agents for territorial security. IEEE Computational Intelligence Magazine 2011;6(1):55–67.

Anderson R. Security engineering: a guide to building dependable distributed systems.John Wiley and Sons; 2001.

Asokan N, Janson P, Steiner M, Waidner M. The state of the art in electronic payment systems. Computer 1997;30:28–35.

Baig ZA, Khan AI. A fault-tolerant scheme for detection of DDoS attack patterns in cluster-based wireless sensor networks. Sensor and AdHoc Networks: Theoretical and Algorithmic Aspects 2008:277–90.

Balaji P, Srinivasan D. Multi-agent system in urban traffic signal control. IEEE Computational Intelligence Magazine 2010;5(4):43–51.
Baqer M, Khan AI, Baig ZA. Implementing a graph neuron array for pattern recognition within unstructured wireless sensor networks. In: Proceedings of the EUC workshops, Lecture notes in computer science, vol. 3823; 2005. p. 208–17.
Bergenti F, Poggi A. Multi-agent systems for E-health: recent projects and initiatives. In: Proceedings of the 10th international workshop on objects and agents; 2009.
Boudaoud K, Labiod H, Boutaba R, Guessoum Z. Network security management with intelligent agents. In: Proceedings of the IEEE/IFIP network operations and management symposium (NOMS); 2000.
Certificate authority, 〈http://en.wikipedia.org/wiki/Certificate_authority〉.
Culler D, Estrin D, Srivastava M. Overview of sensor networks. IEEE Computer 2004:41–9.
Duri S, Elliott J, Gruteser M, Liu X, Moskowitz P, Perez R, Singh M, Tang J-M. Data protection and data sharing in telematics. Journal of Mobile Networks and Applications 2004;9(6):693–701.
Eschenauer L, Gligor VD. A key-management scheme for distributed sensor networks. In: Proceedings of the 9th ACM conference on computer and communications security; 2002.
Fricke S, Bsufka K, Keiser J, Schmidt T, Sesseler R, Albayrak S. Agent-based telematic services and telecom applications. Communications of the ACM 2001;44(4):43–8.
Gligor VD. Security of emergent properties in ad-hoc networks. In: Proceedings of the international workshop on security protocols; 2004.
Guan SU, Hua F. A multi-agent architecture for electronic payment. International Journal of Information Technology and Decision Making 2003;2:497–522.
Jin F, Niu Z, Lang H. A pattern based anti-fraud method in c2c E-commerce environment. In: 2010 international conference on E-business and E-government (ICEE); 2010. p. 242–4.
Kent S, Atkinson R. Ip encapsulating security payload. IETF RFC 2406; 1998.
Khan AI. A peer-to-peer associative memory network for intelligent information systems. In: Proceedings of the thirteenth australasian conference on information systems, vol. 1; 2002.
Khan AI, Mihailescu P. Parallel pattern recognition computations within a wireless sensor network. In: Proceedings of the international conference on pattern recognition, vol. 1; 2004. p. 777–80.
Kolaczek G, Juszczyszyn K. Traffic and attack pattern analysis for multiagent distributed intrusion detection system. In: Proceedings of the international conference on intelligent systems and knowledge engineering; 2007.
Li W, Hoang D. A new security scheme for E-health system. International Journal of Collaborative Technologies and Systems 2009:361–6.
McKnight D, Choudhury V, Kacmar C. The impact of initial customer trust on intentions to transact with a web site: a trust building model. Journal of Strategic Information Systems 2002;11:297–323.
Orfila A, Carbo J, Ribagorda A. Intrusion detection effectiveness improvement by a multiagent system. International Journal of Computer Science and Applications 2005:1–6.
Servin A, Kudenko D. Multi-agent reinforcement learning for intrusion detection. In: Adaptive agents and multi-agents systems; 2007. p. 211–23.
Song R, Korba L. The scalability of a multi-agent system in security services. National Research Council—Canada 2002:1–23.
Stallings W, Brown L. Computer security: principles and practice. Prentice Hall; 2007.
Stoian I, Sanislav T, Capatina D, Miclea L, Valean H, Sz E. Multi-agent and intelligent agents' techniques implemented in a control and supervisory telematic system. In: Proceedings of the automation, quality and testing, robotics; 2006. p. 463–8.
Technologies L. Platforms G.F.I. 21st century territorial security: a sensor network approach, White paper.
Ulieru M, Ionescu D. Privacy and security shield for health information systems. International Journal of Computer Systems Science Engineering 2006;21(3):215–21.
Venkataram P, Babu B, Naveen M, Gungal GS. A method of fraud and intrusion detection for e-payment systems in mobile E-commerce. In: 21st IEEE international performance, computing, and communications conference, 2002; 2007. p. 395–401.
Wang H, Sheng B, Tan CC, Li Q. Comparing symmetric-key and public-key based security schemes in sensor networks: a case study of user access control In: IEEE ICDCS, Beijing, China; 2008.
Weiss G. Multiagent systems: a modern approach to distributed artificial intelligence.The MIT Press; 1999.
Wooldridge M. An introduction to multiagent systems.John Wiley and Sons; 2002.
Ye Y, Yi X. Coalition signature scheme in multi-agent systems. In: Proceedings of the 11th international world wide web conference; 2002.
Zhao S, Liu H, Sun Z. Scalable trust in multi-agent E-commerce system. In: Proceedings of the international symposium on electronic commerce and security; 2008. p. 990–3.