Int. J. Communications, Network and System Sciences, 2016, 9, 1-18
Published Online January 2016 in SciRes. http://www.scirp.org/journal/ijcns
http://dx.doi.org/10.4236/ijcns.2016.91001

A Review of Artificial Immune System Based Security Frameworks for MANET

Lincy Elizebeth Jim, Mark A. Gregory

RMIT University, Melbourne, Australia
Email: lincyek@gmail.com, mark.gregory@rmit.edu.au

Received 16 December 2015; accepted 22 January 2016; published 25 January 2016

Abstract

Mobile ad hoc networks (MANETs) are collections of wireless mobile devices that form a communication network with restricted broadcast range, limited resources and without fixed infrastructure. Routing is a critical function in multi-hop MANETs. At the same time, security in MANETs—especially routing security—presents a number of new and interesting challenges. Communication is achieved by relaying data along routes that are dynamically discovered and maintained through collaboration between the nodes. Advances in the field of artificial immune systems provide an opportunity to improve MANET security and performance. Artificial immune systems mimic the functionality of the human immune system wherein there is clear distinction between self and non self and this delineation is important in a MANET where there is no centralized management. The high level of protection provided to the human body by an evolved immune system can be applied as a security feature in MANET. The current security techniques proposed for MANET have varying degrees of success due to the dynamic nature of a MANET. This paper will review different strategies for the application of artificial immune systems to MANETs.

Keywords

Artificial Immune System, MANET, Security, Routing

1. Introduction

A Mobile Ad hoc Network (MANET) is a collection of wireless mobile devices that form a network without the need for fixed infrastructure except for connections to fixed-wire digital networks. The mobile hosts are not bound to any centralized control facility or system such as base stations or mobile cellular switching centers. Although this offers unrestricted mobility and connectivity to the users, the onus of network management is now entirely a matter for the nodes that form the network. Due to the limited transmission range of many wireless

How to cite this paper: Jim, L.E. and Gregory, M.A. (2016) A Review of Artificial Immune System Based Security Frameworks for MANET. *Int. J. Communications, Network and System Sciences*, **9**, 1-18.
http://dx.doi.org/10.4236/ijcns.2016.91001

mobile devices, multiple network hops may be needed for one node to exchange data with another across the network. In such a mobile network each node operates not only as a host but also as a relay node and router, forwarding packets for other mobile nodes in the network that may not be within the direct wireless transmission range of each other. Each node communicates using an ad hoc routing protocol that allows it to discover multi-hop paths through the network to any other node. The idea of an ad hoc network is sometimes called infrastructure-less networking since the mobile nodes in the network dynamically establish routing among themselves to form their own network on the fly. Routing capability is a basic functionality for multi-hop MANETs where individual nodes act both as host and as a relay or router that forwards packets for nodes that are not in transmission range of destination nodes. Several route discovery algorithms have been proposed in the literature. The concept of ad hoc networks is attractive for the following reasons: a) ease of deployment; b) speed of deployment; and c) decreased dependence on fixed infrastructure. Ad hoc networks have several salient characteristics and features that give rise to the following challenges:

1) Dynamic topologies. Nodes are free to move arbitrarily, thus the network topology, which is typically multi-hop, may change randomly and rapidly at unpredictable times and may consist of both bidirectional and unidirectional links.

2) Limited physical security. Mobile wireless networks are prone to security threats with an increased possibility of eavesdropping, spoofing and denial of service attacks. Existing fixed link security techniques are often applied within wireless networks to reduce security threats. As an inherent benefit the decentralized nature of network control in MANET provides additional robustness against the single point of failure present in centralized approaches.

3) Energy constrained operation. Some or all of the nodes in ad hoc networks may rely on batteries or other intermittent or exhaustible means for their energy. For these nodes an important system design criteria is energy conservation.

Artificial immune system is a branch of artificial intelligence inspired by the principles of invertebrate immune systems. The algorithms in artificial immune system adapt the immune system features of learning and memory to solve a problem. This area of research makes an attempt to bridge the gap between immunology and engineering. Immune system properties are of great interest to engineers and computer scientists. The evolution of AIS has its roots in the work of Farmer, Packard and Perelson [1]. The work carried out by Bersini and Varela [2] was influential and helped in bridging the gap between computing and immunology. Bersini concentrated on how to build models that mimic immune system memory properties. Forrest [3] focussed on network intrusion detection thereby concentrating more on the ability of immune systems to discriminate between self and non-self. These pioneering works led to a great deal of further research on the application of immune inspired approaches to computer security.

This paper is organised into five sections. An insight into MANET and danger theory is explained in Section II. Section III provides a brief introduction to artificial immune systems whilst Section IV provides details on selected AIS algorithms and their applications. Section V inspects work carried out so far in MANET using AIS.

2. Background

MANET is a collection of mobile, decentralized, and self-organized nodes. Securing MANET is a challenge when every node forming the network is a potential threat that could compromise communications using a multitude of approaches.

The Dynamic Source Routing (DSR) [4] protocol uses source routing rather than the hop-by-hop routing used by the majority of other routing protocols, which eliminates the need for frequent route advertisement and neighbor detection packets. Utilizing the concepts and principles of the Human Immune System (HIS) the development of an Artificial Immune System (AIS) for MANET provides an alternative approach to improve network security.

"AIS are intelligent and adaptive systems inspired by the immune system toward real-world problem solving. AIS are adaptive systems inspired by theoretical immunology and observed immune functions, principles and models, which are applied to complex problem domains [5]."

A recent immunological discovery called Danger Theory now paves the way for more efficient, second generation AIS. The Dendritic Cell Algorithm (DCA) [6] is a biologically inspired technique, developed to detect intruders in computer networks. The DCA is based on a metaphor of naturally occurring Dendritic Cells (DCs), a type of cell which is native to the innate arm of the immune system [5]. DCs are responsible for the initial detection of intruders, including bacteria and parasites by responding to the damage caused by the invading entity. Natural DCs receive sensory input in the form of molecules, which can indicate if the tissue is healthy, or in distress. These cells have the ability to combine the various signals from the tissue and to produce their own output signals. The output of DCs instructs the immune system responder cells to deal with the source of the potential damage. DCs are excellent candidate cells for abstraction to network security as they are the body's own intrusion detection agents.

To improve the performance of AIS algorithms a "danger project" has been commenced based mainly on the immunology Danger Theory which states that the response type of the immune system to the incoming pathogens occurs due to the existence of danger or safe signals from the body tissues affected by the pathogen, as illustrated in **Figure 1** [5].

DCA is a danger project contribution that utilizes the DC role in HIS as forensic navigators and important anomaly detectors. DCs are defined as antigens presenting lymphocytes in the innate immunity; these lymphocytes play a key role in either stimulating or suppressing the adaptive immunity T-cells and hence controlling the immune system's response.

DCA's ability to act as an anomaly detector algorithm inspires further investigation of the biological model to introduce improved DC inspired algorithms [6], which could detect other types of security attacks [7] in a MANET. In addition, many of the MANET characteristics and properties are similar to the innate immunity abstract features; such as the openness and susceptibility of each to different types of danger attacks [8].

MANETs share the same basic security goals that occur in other network types. The need for confidentiality, authenticity, integrity, availability, non-repudiation and access control as illustrated in **Figure 2**, which is the same as in other network types [9] and is generally determined by the importance and sensitivity of applications used or data transmitted. Network control, management, and security goals are harder to achieve in a MANET than in conventional networks [10] due to the mobile decentralized nature of the network.

3. Artificial Immune Systems

The field of AIS has achieved importance as a branch of Computational Intelligence since its inception in the 1990s. The four major AIS algorithms on which research is centered are: 1) Negative Selection Algorithm (NSA); 2) Artificial Immune Networks; 3) Clonal Selection Algorithm; and 4) Danger Theory and Dendritic Cell Algorithms. AI brings together the disciplines of Immunology, Computer Science and Engineering. Over the past decade research into the Immune System has gained popularity as a vehicle for novel solutions to complex issues. The highly distributed, adaptive nature of the immune system includes capabilities such as learning, memory and pattern recognition which are solid foundations for an artificial equivalent. AIS outcomes require

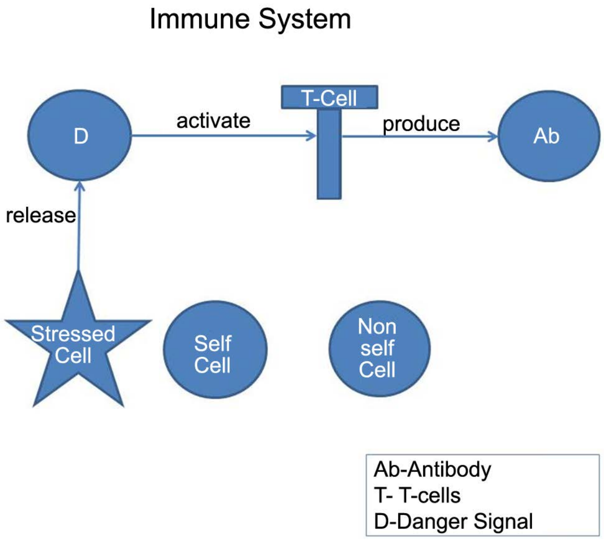

Figure 1. Danger theory model.

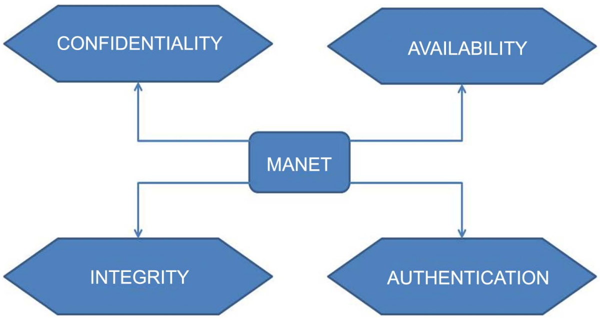

Figure 2. MANET security goals.

both integration of immunology and engineering to transform the complex evolved mechanisms found in the HIS.

Forrest *et al.* [3] proposed a negative selection method to distinguish self from non-self, based on the generation of T-cells. This approach was applied to the problem of virus detection in a computer and raised the profile of negative selection approaches. Following the work by Forrest *et al.* variations of the negative selection algorithm have been developed with the essential properties of the original negative selection algorithm remaining.

Artificial Immune Networks (AINs) are another popular AIS approach that was based on Farmer *et al.*'s immune network model [1] and an early immune network algorithm was designed by Ishida [11]. Timmis *et al.* [12] redefined the artificial immune network. Castro *et al.* [13] proposed the Clonal Selection Algorithm (CLONALG) which is based on a clonal selection principle processes similar to those found in Genetic Algorithms including clone, mutation and reselection.

The AIS community has produced a multifaceted set of immune inspired algorithms in order to solve computational as well as real world problems. Castro and Timmis [14] provided a detailed analysis of the Immune System and a presentation of current AIS algorithms. Tarakanov *et al.* [15] provided an insight into the mathematical basis of immunocomputing. Ishida [16] reviewed immune network models and highlighted the benefits of each approach.

4. AIS Algorithms

4.1. Negative Selection

In the biotic or biological immune system T-cells are initially formed in the bone marrow and on maturation they move to the thymus. The phase of T-cell evolution is characterized by expressions provided by T-cell receptors. Whenever the Pre-T-cells and thymus cells interact this leads to Pre-T-cell multiplication and divergence. Then these T-cells undergo negative selection to eliminate T-cells that are activated by self in the thymus. Although variations of negative selection have been proposed, the process described in [17] [18] still remains in use.

Gao *et al.* [19] proposed enlarging non overlapping detectors to obtain non-self coverage. Let the detector center be O_j, then detector j will have a maximum radius of r_j. The detectors with large radii have higher fitness. Ma *et al.* [20] describe a mechanism to produce useful detectors that are randomly produced and the unmatched antigen is placed into a detector space called the feedback detector. The feedback detector will be eliminated in case it matches self strings. Once the feedback detector becomes mature it will be used to match antigens. When the feedback detector acquires a match on further antigens, it becomes a legitimate detector. Simple Evolutionary NSA (ENSA) and basic ENSA [20] are Negative Selection Algorithm (NSA) variations and the functionality of Simple ENSA is to generate detectors capable of identifying corrupted data. When a detector is to match correct data it can lead to wayward or abnormal changes in the detector and this detector will be discarded. The evolution of the next generation detectors takes place through mutation, positive selection and negative selection. Such evolutionary inception loops to generate detectors continue until a wayward change is detected. In Basic ENSA, in addition to the next generation detector set, a randomly generated detector is also added. By including the additional detector searches can take place in the global space as well. ENSA finds its use in hardware/ software segregation in embedded systems.

The proposed system [21] uses numerous layers to account for fault detection. The negative selection algorithm is implemented in the top most layer in order to identify abnormal situations at the system level. The system identifies non-self states and upon identification of a non-self the cause must be contained. This can only be achieved upon reconfiguration of the device and performing test patterns at a low level. The process used hand selection in order to pick a detector set.

In [2] they have used negative selection to detect anomalies in network traffic. The detectors containing non-self patterns are generated. The TCP packet headers that passed within the intra-LAN as well as between the intra-LAN and external networks were captured and chosen for data sets. The collection of the first data set occurred when there were no intrusions and the remaining data was collected during the simulation of different intrusions such as an IP spoofing attack, guessing rlogin, network hopping attack and a scanning attack. The simulations obtained data that was classified into two categories such as "inter-connection" and "intra-connection".

Inter-connection denoted the connection between internal and external hosts while intra-connection denoted the connection between internal hosts. The result of the experiment resulted in the generation of five different sets of detectors after the negative selection incorporated AIS was run five times. The authors came to a conclusion wherein AIS incorporating negative selection can be used as a filter or screening mechanism for invalid detectors instead of generating competent detectors.

Caldas *et al.* [22] proposed a variant of the negative selection algorithm where a repository database is used to store perceptible indexes of performance for an enterprise. There will be a set of cells known as decision cells which will be responsible for extracting decisions from the repository database and provide feedback about the decision to the repository database. Each decision issue is represented by a decision cell which in turn is composed of x decision receptors. The approach proposed consists of two stages: learning and operation. In the learning stage the decision maker selects the decision cells based on the information in the repository database. These cells constitute the initial reservoir of self-cells, that is a decision cache to be stored in the repository database for later use. In the operation stage the decision creator requests decision cells from the repository database and presents decision related problems to the decision cells for resolution.

The identification of unauthorised access to an organization's computing resource is known as intrusion detection [23]. There can be abuse carried out by authorized users on enterprise resources. The most common detection approach employed is to match a received pattern/signature against a library of patterns/signatures. Each time a match is obtained an alert is generated. There can be high false positive rates with this approach due to the following reasons: i) the speed of data can cause the IDS to miss out on key information; and ii) high noise level on networks due to user accidents, damaged or lost data packets. Although alerts are generated by the IDS, enormous time and resources are consumed. Just like a physician who does not make a diagnosis based on a single symptom of disease, correlation and analysis of multiple alerts are required. The steps carried out are: i) a non-self detector string is created by the generator which tests this string against all known self; and ii) if a match on self occurs, a new string is generated thereby destroying the old string. The new string, after going through the same process if successful, is set to memory type. This interactive system designed and implemented in Java validate the use of a biological system approach towards network problems such as virus elimination and intrusion detection

Graaff *et al.* [24] proposed the Genetic Artificial Immune System (GAIS). Here the counterpart of lymphocyte is known as an artificial lymphocyte. The four states in which an artificial lymphocyte exists are: immature (no priority), mature (medium priority), memory (high priority) and annihilated (low priority). The bit string of an artificial lymphocyte is randomly generated and is made to undergo either positive selection or negative selection. Based on the Hamming distance of the nearest self-pattern to the artificial lymphocyte, it will be assigned a distance threshold value. Whenever a match occurs with a non-self-pattern the Hit counter of the artificial lymphocyte is incremented in order to determine its matching ratio.

GAIS also uses Genetic Algorithms (GA) to evolve artificial lymphocytes. Each artificial lymphocyte is related to a chromosome and the randomly generated artificial lymphocytes will constitute an existing population of a GA.

Amaral *et al.* [25] uses GA to generate a detector in a real valued negative selection algorithm. Every possible detector set is linked to a chromosome. Each gene is considered to be a pointer to a y dimensional detector set. The radius for each detector set is computed by using a decoding function and Monte Carlo integration [26] [27] is used to calculate the volume of the detector set.

4.2. Artificial Immune Networks

Jerne [28] suggested that the immune system is capable of attaining immunological memory due to the presence of B-cells. These B-cells prompt each other as well as restrain connected cells to control over production of B-cells. This is needed to maintain a stable memory.

Hunt and Cooke [29] proposed a system comprising a bone marrow object, a network of B-cell objects and antigen population. Bone marrow objects randomly initialise the B-cell population. The antigen population that is present in the system will be randomly picked and inserted to a point in the B-cell network. Cloning of B-cell objects occur if they are able to bind the antigen population.

Pacheo *et al.* [30] designed an Abstract Immune System Algorithm. There are four strategies necessary for the effectiveness of this model: 1) the affinity between the epitope of an antibody or prototype of an antibody; 2) the restraining of an antibody during epitope recognition; 3) the affinity between antigen and antibody; and 4) the nature of cells to die in the absence of communication. A given antibody type will be prompted or deleted by referring to the recruitment threshold or death threshold.

Omni-aiNet [31] is used to solve singular and multi-objective problems. The advantages identified were: 1) a new grid mechanism to control the spread of a solution in the objective space; 2) adjusting the size of the search space based on a predefined suppression threshold; and 3) axiomatically adapting the investigation of the search space.

Taking advantage of the multi-population property of aiNet, the Multi-objective Multi population Artificial Immune Network (MOM-aiNet) for bi clustering was designed [32]. The advantage of MOM-aiNet is that several sets of non-subjugated solutions are returned in contradiction to a single set of non-subjugated solutions. The subjugation is used to compare the quality of solutions for a given issue, thereby enabling it to measure the solution set given by MOM-aiNet. Out of the data set one row and one column is randomly chosen so that MOM-aiNet produces *y* subpopulation of one bi cluster. In the algorithm for each subpopulation *y* clones subject to the mutation process will be produced. Three steps are involved in mutation which would be randomly chosen with equal probability: 1) delete a row of the column; 2) incorporate a row; and 3) incorporate a column. Whenever the number of non-subjugated elements becomes greater than *y* clones a distance based restraining process occurs so that a small and locally diverse sub-population is maintained.

Stibor *et al.* explored the compression quality of aiNet [33]. Using the Parzen window estimation and Kullback-Leibler divergence a similarity measure between the data set (input) and aiNet dataset (output) was presented. A Parzen window estimator helps find the probability densities of the input and output datasets.

4.3. Clonal Selection

According to the Clonal Selection Theory when the original lymphocyte is activated by binding to the antigen, clonal expansion of the original lymphocyte occurs. During the development of the lymphocyte, if any clone with antigen receptors corresponds to molecules of the organisms own body, it will be eliminated. In the clonal expansion of B-cells the average likeness increased for the antigen that sparked the clonal expansion through likeness maturation. Thus the B-cells are more effectively respond to antigens. Somatic hyper-mutation and the Selective mechanism lead to likeness maturation. Somatic hyper-mutation leads to a miscellany of antibodies by introducing random changes to the genes. Only those genes with a higher accord for the encountered antigen will survive. CLONALG was initially introduced in [34] and described in [13] [35].

Ciccazzo *et al.* [36] proposed a variant of CLONALG termed Elitist Immune Programming (EIP). EIP is an extension of immune programming and the concept of elitism is borrowed from the immune inspired algorithm and is introduced to EIP. A new category of hyper-mutation operators and network based coding is used in EIP. Any hyper-mutation operator can only act on one node or link at a time. This work leads to the proposition of ten ad hoc network based hyper-mutation operators: add-parallel, add-series, delete component, mutate-component-value, copy-component-value, add-random-component, mutate-component-kind, link-modify, shrink and expand-node. The EIP algorithm was applied to a synthesis of topology and sizing of analog electrical circuits. Based on experiments the circuits designed by EIP were an improvement over that achieved using Genetic programming.

Halavati *et al.* [37] included symbiosis to CLONALG and uses relatively specified antibodies which are an approximate solution only as they may not contain the data required. Each antibody will have just one property. Later the algorithm randomly selects an antibody to be included in an assembly. By using repetitive steps an assembly with all of the required properties is built, however, in cases where the algorithm fails to obtain an assembly, antibodies with random values are created for the missing portions and a new assembly is created. The technique of utilizing partially specified antibodies stems from the deduction that a problem can be broken into smaller problems and solutions to these smaller problems may provide an improved overall solution to the overarching problem.

The approach in [38] proposed a variation of CLONALG for software mutation and testing that utilizes the notion of "memory individuals" that lead to the recognition of an antigen rather than using the notion of the CLONALG memory individuals per antigen. An antibody population is initialized with *p* tests either randomly generated or pre-specified. A periodic check is done by the algorithm searching for antibodies that will kill at least one mutant program. A Mutation Store is used to assess the freshness of an antibody and antibodies with a higher similarity score are added to the memory set in order to be returned to the tester. The productiveness of this method was compared against an elitist GA and the results showed that the proposed methodology produces a higher mutation score with lower computational cost.

The Trend Evaluation Algorithm (TEA) proposed by Wilson [39] is similar to CLONALG however it incorporates a long term memory pool as well as short term memory pool by multiplying all of the bound trackers. The processes of Apoptosis and Mutation in the TEA occur across all population members. Consider the case where an antigen *Ag* containing 40 fictional price movements and ten Trends (T1-T10) is built to test the ability of the TEA to identify price trends. Antigen *Ag* is divided into four subsets *Ag*1, *Ag*2, *Ag*3, and *Ag*4. *Ag1* contains two simple trends *T*1 and *T*2 and the more complex trends are involved in *Ag*2, *Ag*3 and *Ag*4. Experiments were done to test the algorithms capability to discern price trends as well as to probe the algorithms influence over the long term memory pool.

4.4. Danger Theory Based

Matzinger, the chief advocate of Danger Theory, proposed this theory in 2002 [40] and highlighted that the "foreignness" of a microbe is not the main factor that ignites a response and "selfness" is no assurance of tolerance. The fundamental idea in Danger Theory states that antigen presenting cells are triggered by danger/alarm signals from sore cells. Danger signals will not be sent by healthy cells or by cells experiencing normal cell death.

Any cell that dies an unnatural death sends out a danger signal and antigens near the dying cell are seized by antigen presenting cells like macrophages and are then presented to the lymphocytes. B-cells also secrete antibodies that match the antigens. The antibodies that match the antigens present in the danger zone will be activated and undergo clonal expansion. Those antibodies that do not match will not be in the danger zone and therefore will not be stimulated.

The Two-Signal Model extended by Bretscher *et al.* [41] explains Danger Theory in a different way where two signals are needed to activate the lymphocytes: 1) antigen recognition; and 2) co-stimulation. Signal 2 indicates that the antigen is threatening.

The Danger Theory has its own disadvantages and Aickelin *et al.* [26] proposed applications of Danger Theory that highlight:

- The presence of an APC is required to present a danger signal.
- A danger signal does not have to be dangerous.
- Danger signals can be positive or negative (presence or absence of signal).
- An estimate of nearness may be used to imitate the danger zone.

Conceptual ideas were also proposed on how the Danger Theory can be used for anomaly detection. Based on the Danger Theory, an immune response is always sparked by danger signals. Low or high memory usage, fraudulent disk activity and so forth could indicate danger signals. The Immune System can react to the antigens in the danger zone once a danger signal is produced. After the dangerous components are identified they are then sent to a special part of the system for further verification. Another application of the Danger Theory used in intrusion detection can be found in [27].

Danger Theory has been applied to data mining problems [26]. Consider the case where a user is browsing a set of documents where each document has a set of attributes. When AIS is implemented the antibodies in the system are used to detect the attributes. Each document browsed by the user will be dispensed to the antibodies. When the user expresses interest in the present document a danger signal is raised and antibodies matching the antigen (attribute in the present document) are triggered and become effecters. Wearisome document attributes will endure the auto reactive antibodies. Finally AIS, learns to become a good filter when searching for enthralling documents.

Prieto *et al.* [42] used a goalkeeper strategy in the Danger Theory Algorithm (DTAL) that takes into account danger signals, lymphocytes and the danger zone. This technique was used in robot soccer, when the ball is on the source side (tissue) an alarm signal (Signal 1) will be triggered. When the ball (antigen) is taken by the opponent to the penalty side (danger zone) Signal 2 will be triggered. When both signals are received the lymphocyte is actuated to clear the ball. This strategy showed a performance above 90%.

The work in [43] highlighted an application of Danger Theory to accentuate the effectiveness of an e-mail classifier system. In web-mining usage of various types of media may cause various signals to be released but in an e-mail system an abnormal email may release a "fascinating" signal of one category. The strong pertinence of these features constitutes a form of the Danger Theory.

4.5. Dendritic Cell Based

The main role of Dendritic Cells (DC) as antigen presenting cells were identified by Steinman and Cohn [44] where DC are comprised of leukocytes which are present in all tissues. They are endowed with a disparate hematopoietic lineage and function in various tissues. Inside various tissues, DC segregate and mature when triggered appropriately; later they relocate to secondary lymphoid tissues where they present antigen to T-cells in order to induce an immune response.

The immature DC occupy body surfaces and are commonly present in an immature state and are unable to stimulate T-cells. Once the foreign pathogens are processed and obtained by the immature DC they migrate to the thymus and the spleen where the immature DC mature and stimulate an immune response. As explained in [45] [46] inflection between the various states of DC is enabled by the recognition of signals including pathogen associated molecular patterns (PAMP), danger signals, apoptotic signals (safe signals) and inflammatory cytokines. These signals are explained as follows:

- PAMPs transform immature DC to mature DC
- Danger signals are given out when a tissue cell is damaged; their strength is lower than PAMPs.
- Safe signals are given out when regulated cell death occurs
- Inflammatory cytokines are given out when general tissue distress occurs and amplify the effect of the other three signals.

The response of the T-cell is determined by the corresponding concentrations of these four types of signals. Semi-mature DCs have a suppressive effect while mature DC have an accentuating effect.

The first Dendritic Cell Algorithm (DCA) was presented by Greensmith *et al.* [47] and it involved combining various signals to investigate the current circumstance of the environment and non-parallel sampling of another data stream (antigen). A fuzzy margin derives in accordance with the concentration of co-stimulatory molecules is an indicator for a DC to stop antigen collection and migrate to a virtual lymph node. The DCA works on the input signals with presumed weights to produce output signals. A value of 1 is assigned if the cumulative mature signal is greater than the cumulative semi-mature signal and vice versa. The mature context presentation of that antigen is calculated relative to the total number of antigens.

The DC is designed as a Libitissue [48] tissue server. There are three stages in this algorithm: initialization, update and aggregation. Initialization deals with setting initial values and the update stage is sub-divided into tissue update and the cell cycle. The Libitissue tissue server comprises the tissue update and cell cycle. Data from the source is given to tissue server through the tissue client. The appearance of new data in the system leads to the provision of input signals for the population of DCs. The cell cycle is a distinct process that occurs at a user defined rate. When all of the antigen data is processed the cell cycle and tissue update process stops. In the final stage aggregation of the collected antigens occurs together with analysis and the Mature Context Antigen Value (MCAV) per antigen is derived.

Gu *et al.* [49] used DCA on the KDD 99 [50] data set after two additional functions were added to the system for optimization: antigen multiplier and a moving time window. The antigen multiplier makes several copies of the antigen, to overcome the problem of "antigen deficiency" that can be given to DCs. In each iteration, new signals are calculated using the moving time window. Based on the results the antigen multiplier and moving time window have equal effect on the DCA using the KD 99 data set.

Oates *et al.* [51] devised a DCA approach for a robot classification problem. Robotic DCA is designed as a stand-alone physiological module for compatibility with comprisal design. The Advanced Robot Interface for Applications (Aria) library's [52] "wander" design is extended with two extra modules: image processing and DCA execution. MCAV coefficients are output by the DCA module approximately once per second. PAMP, safe and danger signals are used as input to the DCA. PAMP originates from the image processing module and the safe signal originates from the Laser Range Finder (LRF). A sonar array having a 360 degree field of view (FOV) is the source of danger signal and the antigen is an integer number which can be uniquely identified by the segment of the test pen. The DCA approach used helped the robot to steer away from obstacles in its path.

The authentic DCA is highly speculative and the Deterministic Dendritic Cell Algorithm (dDCA) [47] attempts to overcome this by using two sets of input signals as well as antigens. The DC is subjected to identical input signals. Here an array is used in order to store the antigen value and count of times the DCs have collected the antigen. There are three parameters in the dDCA-weight scheme for processing signals, outputting DC values and the number of DCs.

The work in [53] depicts the affinity of DCA towards the architecture and operational requirements of sensor networks. Based on this variation, ubiquitous DCA (UDCA) was proposed to detect attacks on sensor networks and its features include:

- Signals from multiple data sources are collected by DC. New output cytokines are accumulated at the maturation stage of each DC.
- Linking of antigens with context information is done by UDCA.
- Extent of node misbehaviour is detected by UDCA via signals generated.

5. MANET and AIS

A fundamental aspect of an ad hoc wireless network is its lack of infrastructure, design issues and challenges stem from this characteristic [54] and the lack of a centralized mechanism adds difficulty by increasing the complexity of fault detection and correction. The dynamically changing nature of mobile nodes causes an unpredictable topology that requires frequent route changes, network partitioning and protection from increased packet loss. The security attacks on MANET networks utilize opportunities provided by the wireless mobile infrastructure in which nodes can join and leave at will using dynamic requests [55]. Energy efficient routing algorithms can be tricked into routing through compromised nodes if the node indicates high power when the other battery powered nodes are showing varying power levels [8]. Failure of one node may affect the entire MANET and this adds to network design complexity especially the probability of network partitioning increases as node power levels fluctuate. Mobile node power supply limitations and energy depletion is a major factor affecting the lifetime of the ad hoc network [56].

HIS has been identified as a source of models, functions, and concepts that inspire AIS algorithms which can be used to secure both host-based and network-based systems [57]. However, it is not only important to utilize the HIS when creating AIS-based algorithms as much as it is important to produce high performance algorithms [5]. Therefore, creating a balance between utilizing HIS and introducing AIS-based intrusion detection algorithms is a crucial issue that would be valuable to investigate because MANET properties raise security issues to a level above those associated with fixed networks. The AIS properties such as being self-healing, self-defensive and self-organizing provide an opportunity to meet the challenges of securing MANET [58].

6. Developments in AIS Based MANET

Sarafijanovic *et al.* [59] investigated the use of AIS to detect node misbehavior in MANET using DSR and the AIS algorithms utilized negative selection and clonal selection. In this proposed system as illustrated in **Figure 3**, each DSR node implements an instance of the detection system, and runs it in two stages. In an initial stage, the detection system learns about the normal behavior of the nodes with respect to the DSR protocol. During this stage, the node is supposed to be in a protected environment in which all nodes behave properly. From the packets received or overheard, the node observes the behavior of its neighbors and creates positive antigens. Towards the end of this learning stage, the node runs the negative selection process and creates its antibodies, known as Detectors.

After the initial stage, the node may leave the protected environment and enter the second stage where detection and classification are done. In this stage, the node may be exposed to misbehaving nodes. The Detectors created in the learning stage are used to check if newly collected antigens represent the behavior of good or bad nodes. In case an antigen, created for any neighbor during some time interval, is detected by any of the Detectors, the neighbor is considered to be doubtful in that time interval. If there are too many doubtful intervals for a neighbor, that neighbor is classified as misbehaving. This triggers the clonal selection process in the node that

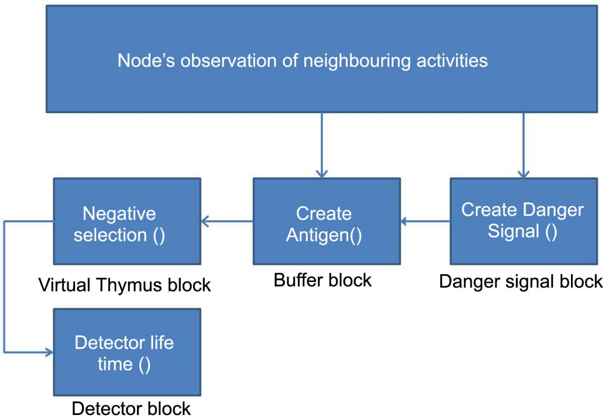

Figure 3. Detection system.

made the classification. In this process, the node adapts its detectors to improve experienced misbehavior detection.

The work carried out by Hoffmeyer *et al.* [60] uses the self-nonself model negative selection process and some form of danger signal. In the system proposed, the Transmission Control Protocol (TCP) connections play the role of self and nonself cells. TCP is a computer networking protocol that provides reliable data packet exchange between two networked devices that communicate over a multi-hop network. One connection is represented by a triplet encoding the sender's destination address, the receiver's destination address, and the receiver's port number. A Detector is a bit sequence of the same length as the triplet and matches a triplet if both have contiguous equal bits. Candidate Detectors are generated randomly; in a learning phase, Detectors that match the correct (*i.e.*, self) triplets are eliminated and this is done offline, by presenting only valid TCP connections. The Detectors that are not eliminated have a finite lifetime and die unless they match a nonself triplet, as in the IS. The danger signal is also used and it is sent by humans as confirmation in case of potential detection. This is a drawback, since human intervention is required to eliminate false positives, but it allows the system to learn changes in the self.

In the implementation of IDS [61] to secure MANETs the authors present an approach based on the paradigm of HIS. This is achieved by using a Mobile Agent which they identify as the Immune Agent (IA). The IA consists of four processes based on the scenarios encountered in the wireless ad hoc domain.

1) Detection process

This is triggered when a connection between two nodes is established.

2) Classification process

The next security process is the classification of self or non-self.

3) Blocking/Isolating Process

The aim of this process is to block and isolate a node which is classified as malicious based on the standards stored in the IA.

4) Recovery Process

The IA takes a snapshot of the data recovery file when it successfully attaches to the new node that intends to join the wireless domain. When a change in the node's system is detected a classification for the pattern that caused the change is also determined. This approach uses memory, where data has been fed into a database and the same data is fetched and used in the recovery process.

Nauman *et al.* [62] proposed using a DC approach in combination with a BEE algorithm. The scouts and foragers of the BEE algorithm are used in the DC formation. This algorithm uses a dynamic detector set and the DCs are modeled to sample the antigens (scouts) from the body tissues (node). During this phase both self and non-self-antigens are sampled. At startup random Detectors are generated which are in turn subjected to negative selection with respect to self-antigens represented by the semi mature DCs.

Using negative selection to generate Detectors involves computational overhead and generating Detectors in a dynamically varying environment like MANET is not feasible.

According to Ye *et al.* [63] two IAs namely the detection agent and counterattack agent are entrusted with detection as well as response. The detecting agent may be viewed as a T-cell lymphocyte while the counterattack agent may be viewed as an antibody. Whenever the detection agent finds an invader, instructions are sent to the counterattack agents. The behavioral patterns of nodes identified are as follows:

- Node Q received message P recorded as Recv (Q, P)
- Node Q sends message P recorded as Send (Q, P)
- Node Q keeps message P recorded as Keep (Q, P)
- Node Q modified message P recorded as Modify (Q, P)
- Node Q deletes message P recorded as Delete (Q, P)
- Node Q generates new message P recorded as Make (Q, P)
- Node Q verifies message P recorded as Verify (Q, P)
- Node Q stores message P recorded as Store (Q, P)
- Node Q broadcasts message P recorded as Broadcast (Q, P)
- Message R is the reply of the message P recorded as Reply (P, R)

The behavior patterns of attack nodes are kept in the Immune Memory Library to represent different attack methods:

- Method 1: Recv (Q, P), Delete (Q, P). Node Q receives the message P and deletes it without transmitting it. This is an Interrupt Attack.
- Method 2: Recv (Q, P), Modify (Q, P), Send (Q, P). Node Q upon receiving the message P modifies it and then transmits it. This is an Error Message Attack.
- Method 3: Recv (Q, P), Reply (P, R), Send (Q, P). Node Q receives message P and sends the message via the wrong route. This is called a Black Hole Attack.
- Method 4: Recv (Q, P), Keep (Q, P), Send (Q, P). Node Q receives message P and then transmits it after keeping the message for some time. This can result in a Hidden Attack.
- Method 5: Make (Q, P), Broadcast (Q, P). Node Q makes and broadcasts a large number of messages in a short time which leads to node overload. This is called a Denial of Service Attack.
- Method 6: Store (Q, P), Modify (Q, P), Send (Q, P). Node Q modifies the details of the route and transmits again which will result in other nodes receiving error filled routing messages.

The detection agent records the behavior of each of the neighboring nodes. When the node behaviors do not match they are analyzed using the Immune Strategy Library.

The creation of the Immune Memory Library and Immune Strategy Library as illustrated in **Figure 4** is not mentioned in detail in [63] and the Detection Agent cannot record the behavior of a particular node as there are other nodes in addition to the neighboring nodes which could be compromised. If the Detection Agent was to record each node's behavior this would result in a considerable computational overhead. This has not been simulated so the accuracy and feasibility of the approach is left to future work.

Fatemeh [64] proposed a combination of AIS and GAs that are used to adapt to changes in network topology and Spherical Detectors are generated to cover non-self-space. The technique used to generate Spherical Detectors is an area for future research that might be used to identify the equivalent to protein compound antigens that exist in body cells alongside pathogens.

The innate immune system uses built in knowledge to combat against infections and a danger signal means damage caused to self-cells due to antigens coming from non-self. In Danger Theory the recognition of pathogens is not enough to get a response from the adaptive immune system but an additional sense of danger is needed before the body reacts to any infection caused by pathogens.

Nauman [65] proposes two approaches based on AIS called BeeAIS and BeeAIS-DC. BeeAIS utilizes negative selection to detect anomalies in MANETs and with the use of negative selection the profile of the system behavior during normal routing is found. The concepts used are:

a) Antigens extracted from incoming traffic in the network are created from the packet header data. Here the antigens are modeled as one of three different types; scout antigen used to detect anomalies relating to scouts and forager antigens of two types used to detect modifications to the source route.
b) Antibodies and Detectors are created by combining four gene values as random numbers.
c) Matching functions are when the interaction between antigen and antibody is evaluated in terms of distance in Hamming shape space.

The two stages of BeeAIS operation are the Learning Phase and the Protection phase. In the Learning Phase the behavior of the system is identified under normal routing conditions where each node monitors traffic in order to collect the data required to create self-antigens. When a scout is received, a node may form a scout anti-

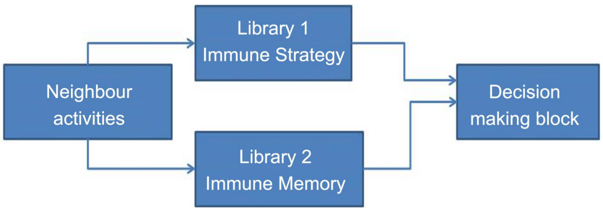

Figure 4. Immune libraries.

gen and when a forager is received; forager antigens are formed. A node could receive the same self-antigen many times. Hence it matches the newly formed antigen with the antigens that have been previously collected from the traffic flow.

After the end of the Learning Phase a set of Detectors are generated using a negative selection process with the self-antigen set collected during the Learning Phase. The Detectors will be generated randomly and only those that do not match with self-antigens are kept.

In the Protection Phase the nodes collect antigens from the incoming traffic and carry out measurement of their affinity with the Detector sets. Whenever a match occurs it indicates an anomaly is present. However this approach fails as the algorithm learns the system behavior only once and as a result during the Protection Phase newly observed behavior is declared as malicious by the system.

Whenever a node receives a forward or backward scout it creates an antigen. After extracting the relevant fields from the scout header an antigen is created. The fields which are extracted include the scout source, destination, length of route and node ID of the previous hop. DCs are formed when a node sees a scout and the DCs are initialized with the following attributes:

a) *DC Antigen*: The sampled antigen from the scout is attached to the DC.
b) *DC Life*: The DCs are assigned a short life and they die a natural death after that.
c) *DC State*: Upon instantiation a DC is an immature DC and when antigens are sampled and when safe signals are present, the DC transitions to a semi-mature state. During the exposure to danger signals DCs transform to mature DCs.

As the system starts, a set of random Detectors are generated by the node and the Detectors have to undergo a negative selection process during which antigens are identified. The Detectors that match with antigens are eliminated and the resulting Detectors are able to match with non-self-antigens. Mature DCs are used to activate T-cells. During matching the T-cell detector is transformed to become an activated Detector.

The approach proposed in [65] doesn't describe the role of an activated Detector sufficiently and the process carried out after the Detector becomes active is not adequately explained. Negative selection requires a Learning Phase which is not practical in a dynamic MANET. How the role of the Detector to curb malicious activities in the MANET is to occur is not adequately explained.

Yasir *et al.* in [61] propose a security approach based on immune inspired properties. Three profiles are created:

1) Gene Profile. This profile would contain the recurring events needed to establish a connection system. This is similar to self-cells in the Immune system.
2) Detector Profile. This profile is used to recognize non-self which is similar to HIS T-cells.
3) Non-Self Profile. This profile would contain events that harm the system.

The Immune agents capture the self and non-self patterns during the monitoring and capturing phase and we learn that $U = S_f \cup N_f$ represents the collection of patterns monitored while packets transfer, it contains both self and non-self patterns. $N_f = \{n_{f1}, n_{f2} \dots n_{fm}\}$, $S_f = \{s_{f1}, s_{f2} \dots s_{fn}\}$ represents the set of all self and non-self patterns captured by the Immune Agent. In order to simulate the T-cells the Immune agent will be equipped with detectors that are randomly generated. $Đ = \{d_1, d_2 \dots d_m\}$ represents the set of the generated Detectors, $Đ' = \{d'_1, d'_2 \dots d'_m\}$ the set of maturated Detectors.

The negative selection algorithm is used to get the maturated Detectors in order to ensure that the generated Detectors do not match any self. The next step is clonal selection where a Detector will be cloned if it attains a score after matching to non-self.

The algorithm is as given below:

1) Let: d'_i score = 0

```
2) For $N_f = \{n_{f1}, n_{f2}… n_{fm}\}$; bind $d'_i$ to $n_{fj}$, (for i, j = 1, 2…, m);
3) If $d'_i$ detects $n_{fj}$, then $d'_i$ score++; end if
4) While $\{d'_i \text{ score} \geq \text{max score}\}$ do clone $d'_i$ // proliferation phase
5) $d'_i = d''_i$;
6) If $d''_i$ match $s_{fi}$; $(1\leq i \geq n)$; then delete $d''_i$ ;// negative selection
7) Else $Ð' = Ð' + d''_i$ // Update the Detectors Profile
```

In order to utilize the Danger Theory concept, the immune agent keeps a replica of data necessary to regain a node. Consider β a system with components at time t: $\beta t = \{\beta_1, \beta_2 ... \beta_n\}$. A copy of β_t is available in the Immune Agent Database. Therefore any change in the system components can be identified. Let ε be a change that occurs in the system after time Δt. The Immune Agent checks the system and observes $\beta t+\Delta t = \beta \pm \varepsilon$; As ε is not recognized it is considered as a suspect pattern and will be included in the non-self-set to be blocked in future. This approach is not implemented and simulated so the accuracy of this approach cannot be validated.

Ansari *et al.* in [66] use the concepts of clonal selection and danger signal for misbehavior detection using DSR. The protocol events of a node are mapped to HIS elements. Genes of a node are designed based on the performance of the network, the node's observations of neighboring nodes. These genes form the basis to detect if a node is misbehaving. Antigens are represented by a pattern of observed events generated by the protocol.

The events generated at the monitored node when it receives a packet originating at sender node are as given below

- i = RREQ sent
- j = RREP sent
- k = RERR sent
- l = DATA sent
- m = RREQ received
- n = RREP received
- o = RERR received
- p = DATA received

The antigen set is represented as:

- D = {mmimmmimmjmimmplplp,plplplplplplplplplp}
- G_1 = Num (m)
- G_2 = Num (m*(i+j))
- G_3 = Num (p)
- G_4 = Num (p*l)

Where G_k denotes the k^{th} gene, Num denotes the number of occurrences, * denotes "zero" or more occurrence. Each bit in the antigen set *D* is termed as a nucleotide. Antibodies are generated randomly after which they are passed for negative selection.

- Ab_1 = {1010100011,0001101100,1100100010,0110001010}
- Ab_2 = {1010111000,0101100100,1010101010,0110101110}
- $Self_{Ag}$ = ({0000100000,0000000100,0000000010},{0000000000,0000000000,0000100000,0000010000}..)

$Self_{Ag}$ denotes self-antigen and the node saves this information when it is in learning phase. Whenever a node experiences packet loss a danger signal is generated. The criteria to realize the self-antigen is not mentioned. The question arises if an antigen is generated by an attacker node in the same time frame. This would result in all nodes considering $Self_{Ag}$ to be a trustworthy pattern and generate antibodies that cannot accurately detect misbehaving nodes. Whenever a "1" occurs in an antibody pattern which matches with the "1" in Self antigen, the antibody will be deleted.

Sarafijanovic *et al.* [67] attempt to detect node misbehavior by making the nodes learn what normal behavior is in a protected environment. In this scenario a self-antigen pattern would be generated and antibody patterns are deleted if there is "1" in every position the antigen has a "1". Here again the question arises if the same antigen pattern could be generated by an attacker node.

In the approach proposed by Barani *et al.* [23] each node extracts a set of feature vectors *y* out of normal network traffic. Each feature vector is represented by a hypersphere with a fixed radius in the feature space. At each time slot $\Delta t_i\ \varepsilon\ t$ every node extracts a *q* dimensional feature vector y_i.

The equation (1) describes the network state.

$$y_i = (y_i^1, y_i^2, \ldots y_i^q) \qquad (1)$$

where y_i^k ε [0, 1] is a measurable feature vector. The feature space is represented by $S_p \subseteq [0.0, 1.0]^p$ where y_i ϵS_p is associated with an antigen.

A feature vector y_i ε S_p at time t is termed normal if it belongs to a normal network state. To generate a set of negative Detectors $N(t)$ every negative Detector $n_j \epsilon N(t)$ is defined as a hypersphere (a_j, bj) where a_j is the centre of the hypersphere and b_j is the radius.

Let $P(t)$ be the set of positive antigens. The Niche NABC algorithm [68] takes $P(t)$ as input and generates a set of $N(t)$ of mature negative detectors. Immature food sources are created so that there will be a minimum overlap with positive antigens. When the quality of food sources cannot be improved further the food source will be abandoned. In this approach there is offline learning phase and online learning phase. The offline learning phase is run in a protected environment and leads to the creation of negative Detectors. This approach does not map a food source to any of the routing or MANET parameters.

Anass *et al.* [69] proposes a detection generation algorithm. In each generation the DCs deliver a set of elements that are of fixed size randomly chosen from the antigens. Based on the context of the element which is presented a number of operations is established to allow memory Detectors to detect intrusive behavior. When the context of the element is dangerous then the algorithm checks if the memory detectors are able to detect the antigen. If the dangerous element is not detected by the memory Detector the algorithm checks if the mature Detectors are able to detect this element. If there is a mature Detector which is able to detect then this mature element is added to the group of memory Detectors. In case the presented element is harmless the algorithm checks if this element is detectable by the memory Detectors in order to remove the corresponding detector.

The context upon which the element is classified to be “dangerous” is not detailed. This experiment is not validated hence it is not possible to verify the authenticity of this approach.

Visconti *et al.* [70] suggests a type 2 fuzzy set based algorithm for detecting misbehaving nodes that is triggered by network danger signals and antigen presenting cells. The person MF [71] approach is used in order to capture the real behavior of a node and the experts provide the Footprint of Uncertainty (FOU). A red region indicates misbehavior of the network pattern, a yellow region indicates suspicious behavior and a white region indicates normal behavior. The binding process invokes the helper T-cells to measure the actual changes v_i of the network parameter f_i and find the region (Red, Yellow, White) to which the Interval type 2 fuzzy parameter is closer. Therefore to conclude if a node is good or bad I2FM is built for the whole network based on all M network parameters. The proposed approach is considered to be a work in progress.

In [72] an immune system approach has been proposed for securing MANET. The Immune Agent consists of three profiles: gene profile, non-self-profile and Detector profile. Gene profile consists of the frequently occurring events for connection establishments. Detector profile is similar to T-cells in the human body for detecting the non-self. The non-self profile contains events that harm the system. The Immune Agent captures and stores the information pertaining to the protocol during the Training phase (Secure) as well as in the insecure phase. The Immune agent will be equipped with Detectors that are randomly generated. The negative selection algorithm ensures that the generated Detectors do not match self. The Detectors that come out of the negative selection stage are cloned whenever they attain a score detecting non-self and the detector profile is updated. This is followed by node in the network sending route request to a node in which Immune Agent (IA) is installed to which the Immune agent installed node sends a reply with a license for IA. The node that generated route request accepts the IA license and establishes connection. The periodic system checker does the monitoring of incoming packets and compares them against earlier stored patterns. If there is no match the pattern the blocking process is called followed by the recovery process.

7. Conclusion

Techniques based on Immunity are becoming more popular and emerging as a new branch of Artificial Intelligence. The negative selection algorithm is being continuously used and modified to help solve problems. This review highlighted that negative selection algorithms are used for new detector generation schemes and a broad discussion is needed between the biologists, computing scientists and engineers to discover new ways of applying AIS. This paper presents a review of the various AIS approaches applied to MANET and discusses how research is tackling the difficult issues surrounding security in a MANET. Most of the approaches identified are either work in progress or have not been validated demonstrating there is scope for further research and new solutions that can be developed and validated.

References

[1] Farmer, J.D., Packard, N.H. and Perelson, A.S. (1986) The Immune System, Adaptation, and Machine Learning. *Physica D*, **22**, 187-204. http://dx.doi.org/10.1016/0167-2789(86)90240-X

[2] Bersini, H. and Varela, F. (1991) Hints for Adaptive Problem Solving Gleaned from Immune Networks. In: Scwefel, H.-P. and Männer, R., Eds., *Parallel Problem Solving from Nature*, *Lecture Notes in Computer Science* Nº 496, Springer Verlag, Berling, 343-354.

[3] Forrest, S., Perelson, A.S., Allen, L. and Cherukuri, R. (1994) Self-Nonself Discrimination in a Computer. *IEEE Symposium on Research in Security and Privacy*, Oakland, 16-18 May 1994, 202-212.
http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=296580&tag=1
http://dx.doi.org/10.1109/risp.1994.296580

[4] Sivakumar, K.A. and Ramkumar, M. (2007) An Efficient Secure Route Discovery Protocol for DSR. *Global Telecommunications Conference*, 2007. *GLOBECOM* '07. *IEEE*, Washington DC, 26-30 November 2007, 458-463.
http://ieeexplore.ieee.org/xpl/articleDetails.jsp?reload=true&arnumber=4411002

[5] Abdelhaq, M., Hassan, R., Ismail, M., Alsaqour, R. and Israf, D. (2011) Detecting Sleep Deprivation Attack over MANET Using a Danger Theory-Based Algorithm. *International Journal on New Computer Architectures and Their Applications* (*IJNCAA*), **1**, 534-541.

[6] Greensmith, J. (2007) The Dendritic Cell Algorithm. Thesis Submitted for the Degree of Doctor of Philosophy, University of Nottingham. http://ima.ac.uk/papers/greensmith_thesis.pdf

[7] Govindan, K. and Mohapatra, P. (2011) Trust Computations and Trust Dynamics in Mobile Adhoc Networks: A Survey. *IEEE Communications Surveys and Tutorials*, **14**, 279-298.
http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5770276

[8] Mohamed, Y. and Abdullah, A. (2012) I2MANET Security Logical Specification Framework. *The International Arab Journal of Information Technology*, **9**, 495-503. http://ccis2k.org/iajit/PDF/vol.9,no.6/1524-1.pdf

[9] Kumari, S. and Shrivastava, M. (2012) Secure DSR Protocol in MANET Using Energy Efficient Intrusion Detection System. *International Journal of Networks and Systems*, **1**, 6-11.
http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=AA56C0ECE603FD33D605E83C804A3F30?doi=10.1.1.380.6348&rep=rep1&type=pdf

[10] Goyal, P., Parmar, V. and Rishi, R. (2011) MANET: Vulnerabilities, Challenges, Attacks and Application. *International Journal of Computational Engineering & Management* (*IJCEM*), **11**, 32-37.
http://ijcem.org/papers12011/12011_17.pdf

[11] Ishida, Y. (1990) Fully Distributed Diagnosis by PDP Learning Algorithm: Towards Immune Network PDP Model. *IEEE International Joint Conference on Neural Networks*, **1**, 777-782. http://dx.doi.org/10.1109/ijcnn.1990.137663

[12] Timmis, J., Neal, M. and Hunt, J. (2000) An Artificial Immune System for Data Analysis. *Biosystems*, **55**, 143-150.
http://dx.doi.org/10.1016/S0303-2647(99)00092-1

[13] De Castro, L.N. and von Zuben, F.J. (2000) The Clonal Selection Algorithm with Engineering Applications. *Genetic and Evolutionary Computation Conference*, *Workshop Proceedings of GECCO*'00, Las Vegas, 8-12 July 2000, 36-37.

[14] Castro, L.N.D. and Timmis, J. (2002) Artificial Immune Systems: A New Computational Intelligence Approach. Springer-Verlag, London.

[15] Tarakanov, A.O., Skormin, V.A. and Sokolova, S.P. (2003) Immunocomputing: Principles and Applications. Springer, New York. http://dx.doi.org/10.1007/978-1-4757-3807-0

[16] Ishida, Y. (2004) Immunity-Based Systems: A Design Perspective. Springer Science & Business Media.
http://dx.doi.org/10.1007/978-3-662-07863-1

[17] Dasgupta, D. (1998) An Overview of Artificial Immune Systems and Their Applications. Springer-Verlag, 3-19.

[18] Levy, G. (2002) Where Numerics Matter: An Introduction to Quasi-Random Numbers. Financial Engineering News.

[19] Gao, X.Z., Ovaska, S.J. and Wang, X. (2006) Genetic Algorithms-Based Detector Generation in Negative Selection Algorithm. *IEEE Mountain Workshop on Adaptive and Learning Systems*, Logan, 24-26 July 2006, 133-137.
http://dx.doi.org/10.1109/SMCALS.2006.250704

[20] Ma, W., Tran, D. and Sharma, D. (2008) Negative Selection with Antigen Feedback in Intrusion Detection. In: Artificial Immune Systems, Springer, 200-209. http://dx.doi.org/10.1007/978-3-540-85072-4_18

[21] Harmer, P.K., Williams, P.D., Gunsch, G.H. and Lamont, G.B. (2002) An Artificial Immune System Architecture for Computer Security Applications. *IEEE Transactions on Evolutionary Computation*, **6**, 252-280.
http://dx.doi.org/10.1109/TEVC.2002.1011540

[22] Caldas, B., Pita, M. and Buarque, F. (2007) How to Obtain Appropriate Executive Decisions Using Artificial Immu-

nologic Systems. 6*th International Conference on Artificial Immune Systems* (*ICARIS* 2007), Santos, 26-29 August 2007, 407-419. http://dx.doi.org/10.1007/978-3-540-73922-7_35

[23] Kim, J. and Bentley, P.J. (2001) An Evaluation of Negative Selection in an Artificial Immune System for Network Intrusion Detection. *Proceedings of the Genetic and Evolutionary Computation Conference* (*GECCO*), 1330-1337.

[24] Graaff, A.J. and Engelbrecht, A.P. (2006) Optimized Coverage of Non-Self with Evolved Lymphocytes in an Artificial Immune System. *International Journal of Computational Intelligence Research* (*IJCIR*), **2**, 127-150. http://dx.doi.org/10.5019/j.ijcir.2006.57

[25] Amaral, J.L.M., Amaral, J.F.M. and Tanscheit, R. (2007) Real-Valued Negative Selection Algorithm with a Quasi-Monte Carlo Genetic Detector Generation. 6*th International Conference on Artificial Immune Systems* (*ICARIS* 2007), Santos, 26-29 August 2007, 156-167. http://dx.doi.org/10.1007/978-3-540-73922-7_14

[26] Aickelin, U. and Cayzer, S. (2002) The Danger Theory and Its Application to Artificial Immune Systems. 1*st International Conference on Artificial Immune Systems* (*ICARIS* 2002), Canterbury, 9-11 September 2002, 141-148.

[27] Greensmith, J., Aickelin, U. and Twycross, J. (2010) Detecting Danger: Applying a Novel Immunological Concept to Intrusion Detection Systems. arXiv preprint arXiv:1002.0696.

[28] Jerne, N.K. (1974) Towards a Network Theory of the Immune System. *Annals of Immunology*, **125C**, 373-389.

[29] Hunt, J.E. and Cooke, D.E. (1996) Learning Using an Artificial Immune System. *Journal of Network and Computer Applications*, **19**, 189-212. http://dx.doi.org/10.1006/jnca.1996.0014

[30] Pacheco, J. and Costa, J.F. (2007) The Abstract Immune System Algorithm. 6*th International Conference on Unconventional Computation*, Kingston, 13-17 August 2007, 137-149. http://dx.doi.org/10.1007/978-3-540-73554-0_14

[31] Coelho, G.P. and Zuben, F.J.V. (2006) Omni-aiNet: An Immune-Inspired Approach for Omni Optimization. 5*th International Conference on Artificial Immune Systems* (*ICARIS* 2006), Oeiras, 4-6 September 2006, 294-308. http://dx.doi.org/10.1007/11823940_23

[32] Coelho, G.P., Franca, F.O.D. and Zuben, F.J.V. (2008) A Multi-Objective Multipopulation Approach for Biclustering. 7*th International Conference on Artificial Immune Systems*, Phuket, 10-13 August 2008, 71-82. http://dx.doi.org/10.1007/978-3-540-85072-4_7

[33] Stibor, T. and Timmis, J. (2007) An Investigation on the Compression Quality of aiNet. *IEEE Symposium on Foundations of Computational Intelligence* (*FOCI* 2007), Honolulu, 1-5 April 2007, 495-502. http://dx.doi.org/10.1109/FOCI.2007.371518

[34] Cutello, V., Narzisi, G., Nicosia, G. and Pavone, M. (2005) Clonal Selection Algorithms: A Comparative Case Study Using Effective Mutation Potentials. In: Jacob, C., Pilat, M., Bentley, P. and Timmis, J., Eds., A*rtificial Immune Systems*, Vol. 3627, Springer Berlin Heidelberg, 13-28. http://link.springer.com/chapter/10.1007%2F11536444_2#

[35] De Castro, L.N. and von Zuben, F.J. (2002) Learning and Optimization Using the Clonal Selection Principle. *IEEE Transactions on Evolutionary Computation*, **6**, 239-251. http://dx.doi.org/10.1109/TEVC.2002.1011539

[36] Ciccazzo, A., Conca, P., Nicosia, G. and Stracquadanio, G. (2008) An Advanced Clonal Selection Algorithm with Ad Hoc Network-Based Hyper Mutation Operators for Synthesis of Topology and Sizing of Analog Electrical Circuits. 7*th International Conference on Artificial Immune Systems*, Phuket, 10-13 August 2008, 60-70. http://dx.doi.org/10.1007/978-3-540-85072-4_6

[37] Halavati, R., Shouraki, S.B., Heravi, M.J. and Jashmi, B.J. (2007) An Artificial Immune System with Partially Specified Antibodies. 9*th Annual Conference on Genetic and Evolutionary Computation* (*GECCO* 2007), London, 7-11 July 2007, 57-62. http://dx.doi.org/10.1145/1276958.1276967

[38] May, P., Timmis, J. and Mander, K. (2007) Immune and Evolutionary Approaches to Software Mutation Testing. 6*th International Conference on Artificial Immune Systems* (*ICARIS* 2007), Santos, 26-29 August 2007, 336-347. http://dx.doi.org/10.1007/978-3-540-73922-7_29

[39] Wilson, W.O., Birkin, P. and Aickelin, U. (2006) Price Trackers Inspired by Immune Memory. 5*th International Conference on Artificial Immune Systems* (*ICARIS* 2006), Oeiras, 4-6 September 2006, 362-375. http://dx.doi.org/10.1007/11823940_28

[40] Matzinger, P. (2002) The Danger Model: A Renewed Sense of Self. *Science*, **296**, 301-305. http://dx.doi.org/10.1126/science.1071059

[41] Bretscher, P. and Cohn, M. (1970) A Theory of Self-Non Self Discrimination. *Science*, **169**, 1042-1049. http://dx.doi.org/10.1126/science.169.3950.1042

[42] Prieto, C.E., Nino, F. and Quintana, G. (2008) A Goalkeeper Strategy in Robot Soccer Based on Danger Theory. *IEEE Congress on Evolutionary Computation*, Hong Kong, 1-6 June 2008, 3443-3447. http://dx.doi.org/10.1109/cec.2008.4631263

[43] Secker, A., Freitas, A. and Timmis, J. (2005) Towards a Danger Theory Inspired Artificial Immune System for Web

Mining. In: Scime, A., Ed., *Web Mining*: *Applications and Techniques*, Idea Group, Hershey, 145-168. http://dx.doi.org/10.4018/978-1-59140-414-9.ch007

[44] Steinman, R. and Cohn, Z. (1973) Identification of a Novel Cell Type in Peripheral Lymphoid Organs Mice. *The Journal of Experimental Medicine*, **137**, 1142-1162. http://dx.doi.org/10.1084/jem.137.5.1142

[45] Kapsenberg, M.L. (2003) Dendritic-Cell Control of Pathogen-Driven T-Cell Polarization. *Nature Reviews Immunology*, **3**, 984-993. http://dx.doi.org/10.1038/nri1246

[46] Jamie, T. and Aickelin, U. (2004) Towards a Conceptual Framework for Innate Immunity. 3*rd International Conference on Artificial Immune Systems* (*ICARIS* 2004), Catania, 13-16 September 2004, 112-125.

[47] Greensmith, J. and Aickelin, U. (2008) The Deterministic Dendritic Cell Algorithm. 7*th International Conference on Artificial Immune Systems*, Phuket, 10-13 August 2008, 291-302. http://dx.doi.org/10.1007/978-3-540-85072-4_26

[48] Greensmith, J., Aickelin, U. and Twycross, J. (2006) Articulation and Clarification of the Dendritic Cell Algorithm. 5*th International Conference on Artificial Immune Systems* (*ICARIS* 2006), Oeiras, 4-6 September 2006, 404-417. http://dx.doi.org/10.1007/11823940_31

[49] Gu, F., Greensmith, J. and Aickelin, U. (2008) Further Exploration of the Dendritic Cell Algorithm: Antigen Multiplier and Time Windows. 7*th International Conference on Artificial Immune Systems*, Phuket, 10-13 August 2008, 142-153. http://dx.doi.org/10.1007/978-3-540-85072-4_13

[50] Güneş Kayacık, H., NurZincir-Heywood, A. and Heywood, M.I. (2005) Selecting Features for Intrusion Detection: A Feature Relevance Analysis on KDD 99 Intrusion Detection Datasets.

[51] Oates, R., Greensmith, J., Aickelin, U., Garibaldi, J. and Kendall, G. (2007) The Application of a Dendritic Cell Algorithm to a Robotic Classifier. 6*th International Conference on Artificial Immune Systems* (*ICARIS* 2007), Santos, 26-29 August 2007, 204-215. http://dx.doi.org/10.1007/978-3-540-73922-7_18

[52] http://users.isy.liu.se/en/rt/andrecb/fidodido/doc/AriaReference.pdf

[53] Kim, J., Bentley, P., Wallenta, C., Ahmed, M. and Hailes, S. (2006) Danger Is Ubiquitous: Detecting Malicious Activities in Sensor Networks Using the Dendritic Cell Algorithm. 5*th International Conference on Artificial Immune Systems* (*ICARIS* 2006), Oeiras, 4-6 September 2006, 390-403. http://dx.doi.org/10.1007/11823940_30

[54] Bang, A.O. and Ramteke P.L. (2013) MANET: History, Challenges and Applications. *International Journal of Application or Innovation in Engineering & Management* (*IJAIEM*), **2**, 249-251. http://ijaiem.org/volume2issue9/IJAIEM-2013-09-27-063.pdf

[55] Akbani, R. (2009) Defending Against Malicious Nodes in Closed MANETs through Packet Authentication and a Hybrid Trust Management System. PhD Thesis, The University of Texas at San Antonio, San Antonio.

[56] Kargl, F., Geiß, A. and Schlott, S. (2005) Secure Dynamic Source Routing.

[57] Gu, F., Greensmith, J. and Aicklein, U. (2012) The Dendritic Cell Algorithm for Intrusion Detection.

[58] Abdelhaq, M., Hassan, R., Ismail, M. and Israf, D. (2011) Detecting Resource Consumption Attack over MANET Using an Artificial Immune Algorithm. *Research Journal of Applied Sciences*, *Engineering and Technology*, **3**, 1026-1033.

[59] Sarafijanovic, S. and Le Boudec, J.-Y. (2004) An Artificial Immune System for Misbehaviour Detection in Mobile Ad Hoc Networks with Virtual Thymus, Clustering, Danger Signal, and Memory Detectors. *ICARIS* 2004, Catania, 13-16 September 2004, 342-356.

[60] Hofmeyr, S. and Forrest, S. (2000) Architecture for an Artificial Immune System. *Evolutionary Computation*, **7**, 45-68. http://dx.doi.org/10.1162/106365600568257

[61] Mohamed, Y.A. and Abdullah, A.B. (2010) Implementation of IDS with Response for Securing MANETs. 2010 *International Symposium in Information Technology* (*ITSim*), Kuala Lumpur, 15-17 June 2010, 660-665. http://dx.doi.org/10.1109/ITSIM.2010.5561608

[62] Mazhar, N. and Farooq, M. (2008) A Sense of Danger: Dendritic Cells Inspired Artificial Immune System (AIS) for MANET Security. *GECCO*'08, Atlanta, 12-16 July 2008, 63-70. http://dx.doi.org/10.1145/1389095.1389105

[63] Ye, X. and Li, J.S. (2010) A Security Architecture Based on Immune Agents for MANET. *International Conference on Wireless Communication and Sensor Computing*, *ICWCSC* 2010, Chennai, 2-4 January 2010, 1-5.

[64] Barani, F. (2014) A Hybrid Approach for Dynamic Intrusion Detection in Ad Hoc Networks Using Genetic Algorithm and Artificial Immune System. 2014 *Iranian Conference on Intelligent Systems* (*ICIS*), Bam, 4-6 February 2014, 1-6. http://dx.doi.org/10.1109/iraniancis.2014.6802607

[65] Mazhar, N. (2010) Energy Efficient Security in MANETs: A Comparison of Cryptographic and Artificial Immune Systems. *Pakistan Journal of Engineering and Applied Sciences*, **7**, 71.

[66] Ansari, M.S.A. and Inamullah, M. (2011) Misbehavior Detection in Mobile Ad Hoc Networks Using Artificial Im-

mune System Approach. 2011 5*th IEEE Conference on Advanced Networks and Telecommunication Systems* (*ANTS*), Bangalore, 18-21 December 2011, 1-6. http://dx.doi.org/10.1109/ANTS.2011.6163683

[67] Sarafijanovic, S. and Le Boudec, J.-Y., (2005) An Artificial Immune System Approach with Secondary Response for Misbehavior Detection in Mobile Ad Hoc Networks. *IEEE Transactions on Neural Networks*, **16**, 1076-1087. http://dx.doi.org/10.1109/TNN.2005.853419

[68] Barani, F. and Abadi, M. (2011) An ABC-AIS Hybrid Approach to Dynamic Anomaly Detection in AODV-Based MANETs. 2011 *IEEE* 10*th International Conference on Trust*, *Security and Privacy in Computing and Communications* (*TrustCom*), Changsha, 16-18 November 2011, 714-720. http://dx.doi.org/10.1109/TrustCom.2011.92

[69] Khanous, A., Rghioui, A., Elouaai, F. and Bouhorma, M. (2014) MANET Security: An Intrusion Detection System Based on the Combination of Negative Selection and Danger Theory Concepts. 5*th International Conference on Next Generation Networks and Services* (*NGNS*), Casablanca, 28-30 May 2014, 88-91. http://dx.doi.org/10.1109/ngns.2014.6990233

[70] Visconti, A. and Tahayori, H. (2009) Detecting Misbehaving Nodes in MANET with an Artificial Immune System Based on Type-2 Fuzzy Sets. *International Conference for Internet Technology and Secured Transactions*, *ICITST* 2009, London, 9-12 November 2009, 1-2. http://dx.doi.org/10.1109/ICITST.2009.5402588

[71] Mendel, J.M. (2007) Computing with Words and Its Relationships with Fuzzistics. *Information Sciences*, **177**, 988-1006. http://dx.doi.org/10.1016/j.ins.2006.06.008

[72] Mohamed, Y.A. and Abdullah, A.B. (2009) Immune-Inspired Framework for Securing Hybrid MANET. *IEEE Symposium on Industrial Electronics and Applications*, *ISIEA*, Kuala Lumpur, 4-6 October 2009, 301-306. http://dx.doi.org/10.1109/isiea.2009.5356451