Artificial Immune Systems: Survey and Applications in Ad Hoc Wireless Networks

Martin Drozda and Helena Szczerbicka
University of Hannover, Department of Computer Science,
FG Simulation und Modellierung, Welfengarten 1, 30167 Hannover, Germany.
Email: {drozda,hsz}@sim.uni-hannover.de

Keywords: wireless ad hoc network, misbehavior detection, artificial immune system.

Abstract

This document reviews recent efforts in the area of Artificial immune systems (AIS) and their applications for (ad hoc) wireless networks. It presents basic mechanism of Human immune systems, introduces the reader to the learning paradigms of AIS, sums up misbehavior in ad hoc wireless networks and discusses pros and cons of AIS in increasing robustness of ad hoc wireless networks against misbehavior.

INTRODUCTION AND MOTIVATION

Ad hoc wireless networks lack a centralized authority that controls the flow of packets. Instead, each node (mobile device) in an ad hoc network[1] serves as a router. Each node is able to forward packets only to its neighbors[2], and vice-versa each node is able to receive packets only from its neighbors. Nodes are allowed to move and can be switched off and on at any time. Due to the lack of a centralized authority ad hoc networks are extremely vulnerable to user misbehavior. Since nodes within an ad hoc network are expected to have limited computational power and be battery powered, a system that is going to protect them has to be *lightweight*. Additionally, it has to be adaptive as ad hoc networks are expected to operate autonomously with sporadic maintenance [33]. Therefore classical intrusion detection approaches, many of which are based on intrusion signatures, are not suitable for this task.

An example of systems that fulfill the above requirements are Artificial immune systems (AIS). AIS are based on a mechanism that is present in human bodies, namely, on *the Human immune system (HIS)*; see [17, 22, 46, 55, 4] and references therein. AIS are a part of recent promising advances in Intrusion detection systems [39, 54, 52, 42, 14, 5, 48, 56, 38].

ARTIFICIAL IMMUNE SYSTEMS

Background

The Human immune system is a rather complicated mechanism that is able to protect humans against an amazing set of extraneous attacks. This system is remarkably efficient, most of the time, in discriminating between *self* and *non-self* antigens.[3] A non-self antigen is anything that can initiate an immune response; examples are a virus, bacteria, or splinter. The opposite to non-self antigens are self antigens; self antigens are human organism's own cells.

The important features of HIS have often a dual nature. These dual natures include self vs non-self recognition, innate vs acquired immunity, primary vs secondary response, or general vs specific response. Some immunity mechanisms are antigen specific, systemic (not confined to a local area), or have memory (they are able to launch a stronger response next time a specific antigen is encountered). Often are certain phenomena explained only through hypotheses, an example is the theory of idiotypic networks by Jerne [29]; this theory attempts to characterize the dynamics of interaction between antibodies and antigens.

The above mentioned mechanisms are a result of complex chemical and biological reactions within our bodies. These reactions employ different kinds of cells, proteins, or molecules. Examples are B- and T-cells, macrophages, dendritic cells, killer cells, mast cells, interleukins, interferons etc. These cells act in a distributed manner at various places in a human body such as bone marrow, tonsils, thymus, adenoids, Peyer's patches, or the appendix.

Learning

The process of T-cells maturation in thymus is used as an inspiration for learning in AIS. T-cells are covered by receptors that are able to bind antigens. The creation of T-cells (detectors) in thymus is a result of a pseudo-random process. After a T-cell is created (see Figure 1), it undergoes a censoring process called *negative selection*. During negative selection T-cells that bind self are destroyed. Remaining T-cells are introduced into the body. The recognition of non-self is then done by simply comparing T-cells that survived negative selection with a suspected non-self. This process is depicted in Figure 2. It is possible that the self set is incomplete, while a T-cell matures (tolerization period) in the thymus. This leads to producing T-cells that should have been removed from the thymus and can cause an autoimmune reaction, i.e. it leads to *false positives*. After being introduced into the body, T-cells

[1] We will use ad hoc wireless network, ad hoc network, or simply just network interchangeably.

[2] Nodes that lie within radio range of the sending node.

[3] Self and non-self in short.

divide and die off slowly, maintaining a homeostatic number[4] of T-cells. Only when an antigen enters the body, this homeostatic number changes. There are several basic types of T-cells. Killer T-cells are able to initiate cell lysis (cell dissolution). Helper T-cells orchestrate the immune response; they can also activate a nearby B-cell to produce antibody. The role of memory T-cells is to response more effectively to recurring pathogens (stronger secondary response).

Similar to the negative selection process of T-cells is the *positive selection* process of B-cells. B-cells are produced in bone marrow. Those that bind a non-self antigen are allowed to mature and undergo clonal selection. Some of these B-cells become plasma cells that produce antibodies (each B-cell produces a specific antibody) and some become memory B-cells.

A usual immune mechanism can be concisely described as follow:

1. After the first line of defense (e.g. skin) failed, an antigen enters the human body. It is immediately engulfed by a macrophage (or eating cell) that processes this antigen and displays his pieces on its surface.
2. Helper and killer T-cells are activated by antigen presenting macrophage, if a T-cell recognizes this specific antigen.
3. Helper T-cells activate B-cells. These B-cells undergo clonal selection and start producing antibodies that can bind to the specific antigen. Antibodies efficiently tag antigens and inactivate them by complement fixation (cell lysis), neutralization (binding to specific sites to prevent attachment by an antigen), agglutination (clumping), precipitation, etc. B-cells that get activated more often become memory B-cells. These cells help to respond more efficiently when infection by that kind of antigen re-occurs.
4. Helper and killer T-cells replicate, some of them become memory T-cells that help to launch a faster response next time the same antigen is encountered. Killer T-cells are activated by helper T-cells; activated killer T-cells destroy antigen.

Humans are already born with a "pre-designed" set of cells, proteins and molecules. This is a part of the innate immunity. This innate immunity is later extended by acquired immunity.

Vaccination stimulates the future immune response. Vaccination means that (weakened) antigens are artificially introduced into the body. A usual immune response is triggered, thus producing memory B- and T-cells that stay there for many years, ready to react when the same or similar antigen enter the body in the future.

For further details on human immunology we refer the interested reader to classical texts such as [28]. We would like to note that the central mechanism within human immunology is the *ability to discriminate between self and non-self*. Restated it means that it is possible to distinguish between cells that are not harmful to human body and cells that have the affinity for causing harm.

[4]The number of T-cells stays in a (near) equilibrium state. An antigen can activate T-cells and upset this equilibrium.

Modeling of Negative Selection

Detectors[5] (and antigens) are often represented as strings over an m-ary alphabet, where m usually equals 2, i.e. they are represented as binary strings. Detectors represented as binary strings fall in one of these two categories. They are either represented as plain strings [11] or as binary strings with an attached recognition radius [18]. According to [20], if detectors are represented as binary strings, (partial) matching rules for the censoring process can be divided into three basic categories: statistical, binary distance and landscape affinity. Correlation coefficient is an example of the first category. Hamming distance, r-contiguous bits are examples of the second category. In landscape affinity rules are detectors (antigens) represented as a skyline curve or a landscape; an example of a landscape matching rule is the slope-matching rule. Each of these matching rules attempt to accounts for imperfect matches between detectors and antigens; this is motivated by the underlying mechanisms of HISs.

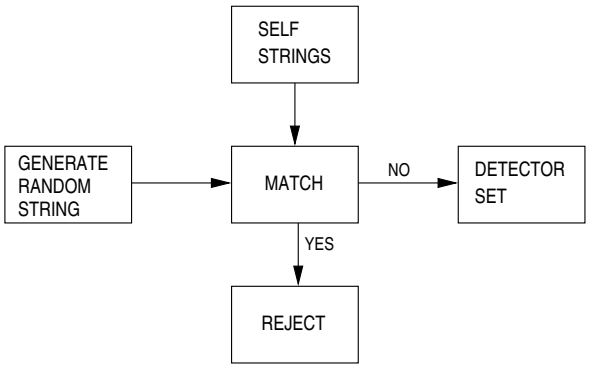

Figure 1. Negative selection.

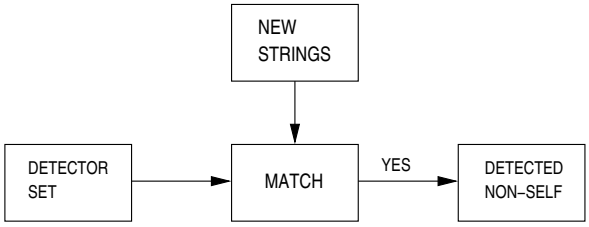

Figure 2. Non-self detection.

[5]We will use terms detector and T-cell interchangeably.

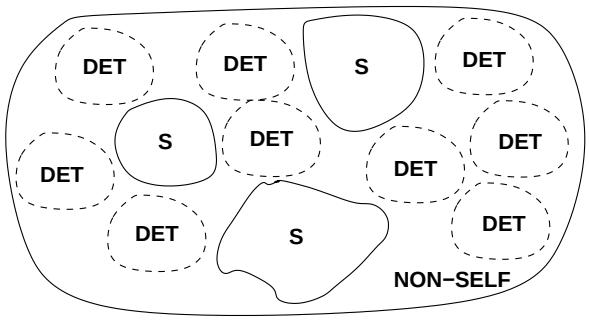

Figure 3. The universe is partitioned into two sets: self and non-self. Non-self should be completely covered by detectors but this is usually not the case, due to existing holes. S = Self, DET = Detector.

The r-contiguous bits matching rule is by far the most popular measure. Two bit-strings (of the same length) match under the r-contiguous matching rule if there exists a substring of length r at position p in each of them and these substrings are identical. It has been thoroughly analyzed and simplified to the r-chunks matching rule in [15, 16].

A detector d with a real valued recognition radius r_{ns} is represented by a tuple (c_d, r_{ns}), $c_d \in [0,1]^n$, $r_{ns} \in R$, where c_d is the center of the detector and r_{ns} is the non-self recognition radius. An element e lies within the recognition radius of d if $dist(d,e) < r_{ns}$, where the distance measure is the Euclidean distance.

Pioneering work in proposing efficient algorithms for detector creation was done by P. D'haeseleer in [11, 10]. An efficient negative selection algorithm maximizes coverage of the non-self set and minimizes the number of detectors needed for such a task. This implies that two detectors should not overlap, if possible; this is however not guaranteed when detectors are produced in a pseudo-random way as depicted in Figure 1. Two algorithms based on dynamic programming were proposed: a linear time algorithm with time and space complexity $O((l-r)N_S) + O((l-r)2^r) + O(lN_R)$ and $O((l-r)^2 2^r)$, respectively, and a greedy algorithm with time and space complexity $O((l-r)2^r N_R)$ and $O((l-r)^2 2^r)$, respectively, where l is non-self string length, r specifies the number of bits that have to match under the r-contiguous bits matching rule, N_R is the desired detector set size, N_S is the self set size. The estimate for N_R, the desired size of the detectors set, is $2^r(1-2^{-r})^{N_S}$. The disadvantage of these and all other negative selection algorithms with a fixed matching rule is that they introduce *holes*, i.e. areas of non-self that it is not possible to detect; see Figure 3. According to [11], holes can be eliminated with adaptive matching rules that produce detectors with high specificity. The greedy algorithm was extended to m-ary alphabet strings in [47]; additionally, the author evaluated pros and cons of representing detectors and antigens with an alphabet of arity $m = 2$ and $m > 2$. The author concludes that both options are justified dependent on the nature of anomaly that should get detected.

In [15] a formal framework for positive and negative selection schemes has been proposed. The framework aims at analyzing these schemes in terms of the number of detectors needed to cover the self or non-self set, respectively. The authors further introduce a new matching rule (r-chunks matching rule).

In order to overcome the complexity of negative selection algorithms based on a bit-string matching rule, in [30] the authors proposed a real-valued negative selection algorithm. Under this algorithm the center of a detector is randomly chosen and the recognition radius is grown until it comes in contact with a self element. In [50] a comparison of the above real-valued negative selection algorithm with three other approaches is undertaken. The comparison is done on a data set known to include traces of misbehavior. Authors conclude that under their settings, the real-valued algorithm failed to dominate other techniques.

The current AIS are usually based on the negative selection mechanism. Certain aspects of the positive selection in terms of percolation theory[6] are discussed in [21]; recognition radius and shape of detectors, and their impact on the number of detectors is studied.

AD HOC WIRELESS NETWORKS

The paradigm of ad hoc wireless networks is *connectivity anywhere, at any time, without any fixed infrastructure*.

The paradigm of ad hoc networking is often restated in graph theoretic framework as follows: an ad hoc network is a net $N = (n(t), e(t))$ where $n(t), e(t)$ are the set of nodes and edges at time t, respectively. Nodes correspond to mobile users or automated sensors that wish to communicate with each other. An edge between two nodes A and B is said to exist when A is within the radio transmission range of B and vice versa. The imposed symmetry of edges is a usual assumption of many mainstream protocols. The change in the cardinality of sets $n(t), e(t)$ can be caused by the freedom that users have when they wish to switch on or switch off their communication device, or can be caused by mobility of users, signal propagation, link reliability and other factors. Data exchange in a point-to-point (uni-cast) scenario usually proceeds as follows: a user initiated data exchange leads to a route query at the network layer of the OSI stack. A routing protocol at that layer attempts to find a route to the data exchange destination. This request may result in a path of non-unit length. This means that a data packet in order to reach the destination has to rely on successive forwarding by intermediate nodes on the path. Therefore the ability to adapt routing when necessary in order to transmit data is another key feature of ad hoc networks.

[6] Percolation theory deals with the effects of varying number of interconnections in a random network.

Battery power that is necessary at each node for reception or transmission of data packets, and for all necessary computation as prescribed by different protocols is of rare nature and therefore its preservation is an important requirement. We will assume, for the sake of this review, that the primary source of electric power for nodes are batteries. The consequences of this assumption are that computation at nodes should be kept to a minimum; any data structure that is implemented at any node is subject to space restrictions. Furthermore, reception and forwarding of "unsolicited" packets should be subject to monitoring and, possibly, to a corrective action.

Protocols at any level of the OSI stack, suitable for ad hoc networking, are reviewed in standard textbooks and other documents such as [41, 43, 26]. Therefore we will not discuss peculiarities of individual protocols and their performance in scope of ad hoc wireless networks.

Performance of ad hoc networks is usually measured in terms of Quality of Service (QoS) parameters. Basic QoS parameters are end-to-end packet delay, number of packets received, long and short term fairness, and overhead at any level of the OSI stack. Other QoS parameters include overhead at different layers of the OSI stack[7], spatial use of control packets, and a multitude of other parameters that are often specific to a given protocol.

Even though ad hoc networks are to some extent robust to misbehavior of single nodes, it makes sense to provide them with features enhancing their survivability. Survivability is defined as *the capability of a system to fulfill its mission in a timely manner, even in the presence of attacks, failures or accidents* [36].

MISBEHAVIOR IN (AD HOC) WIRELESS NETWORKS

In this section we review a few known types of misbehavior that can lead to decreased Quality of Service in wireless networks.[8] They can be classified as Byzantine misbehavior, impersonification and lying, denial of service, selfish behavior, and openly malicious behavior. We note that solutions to some of these attacks have been already proposed. We would like to bring to the reader's attention that packet traces with anomalous behavior can be found at ArachNIDS [1][9]; these can be used for testing and training of an intrusion detection system.

We focus on misbehavior at MAC, routing and transport layers. We assume that the limited battery power makes misbehavior evaluation at higher layers prohibitively expensive. We also assume that misbehavior at the physical layer is neglectable.

[7]For example RREQ, RREP or RERR control packets at routing layer; RTS, CTS, ACK at MAC layer; number of back-offs at MAC layer, etc.

[8]We include attacks that are known from wireless networks that can also be launched in ad hoc networks.

[9]The packet traces are mostly for wired networks.

Link (MAC) Layer:

Medium access selfishness. A selfish node will try to keep the medium busy in order to gain an unshared access to it. This can be done through manipulation of the Network allocation vector in 802.11 class of protocols, through decreasing the size of interframe spaces, or through back-off manipulation; see [9, 35, 8, 19].

Receiver misbehavior. The receiver does not respond to senders RTS's under this scenario, or it can add a large delay penalty to chosen senders; see [35].

Network (routing) Layer:

Overloading. In overloading attacks an attacker injects messages that he knows are invalid. These will be detected and filtered-out but will also be computationally very demanding. It will put the attacked host into a busy-trashing mode; see [27].

Manipulation of routing tables, route caches, and data structures with routing information. An attack aimed at originating inconsistencies in network and creating collisions; see [45, 54]. Ref. [51] discusses an interesting manipulation by creating bogus RREP packets. Ref. [40] discusses a possibility of advertising routes that a given node cannot serve. Another possibility is injecting a RREQ packet with a high sequence number; this will cause that all other legitimate RREQ packets with lower sequence number will be deleted.

Wormholes can exist when two attackers are linked by a private high-speed connection. Any packet to be forwarded is first sent over this private link. This can potentially distort the topology, and attackers may be able to create a virtual vertex cut that they control; see [25].

Gratuitous detour. In this scenario an attacker will try to make the routes through itself to appear longer by appending virtual nodes to found legitimate routes; see [23].

Black and gray holes are created by an attacker or more attackers in order to attract traffic into them and subsequently drop all or selected packets; see [23, 2].

Rushing attacks were introduced in [24]; in routing protocols that utilize the RREQ-RREP handshake it is customary that only the first RREQ packet is forwarded by a given node. Thus a node that manages to forward a RREP packet as the first one, will most likely be included in a forwarding route. This attack can be combined with dynamic power level control or wormholes.

Impersonation or IP spoofing is performed by introducing packets that have stated originators different from real; see [45].

Packet forwarding misbehavior is usually understood as packet dropping, packet duplicating, and packet jamming. It can be partially eliminated by the Watchdog technique [37];

the assumptions are that the given hardware device is able to function in promiscuous mode and that power level control and directional antennas are not used.

Sybil attack is done by creating a number of fictitious nodes; see [12].

Transport Layer:

Selfish misbehavior. Under this scenario the sender ignores rules for congestion window adjustment. It tries to set the congestion window to a maximum size in order to increase his throughput.

TCP SYN flooding aims to exploit vulnerability of a host when a TCP connection is half-open. Under this scenario, a client attempts to connect to a host, leaves however the connections half-open, and continues with opening other connections. The connection buffer of the host overflows; legitimate connections is not possible to open anymore; see [32].

ACK division, DupACK spoofing, and optimistic ACKing. This misbehavior is aimed at manipulation of the size of the congestion window at senders; see [44].

JellyFish attacks. Introduced in [2], they target the congestion control of TCP-like protocols. These attacks obey all the rules of TCP, nevertheless, they are very damaging. Three kinds of JellyFish (JF) attacks were discussed in [2]: JF reorder attack, JF periodic dropping attack, and JF delay variance attack.

APPLICATIONS FOR (AD HOC) WIRELESS NETWORKS

Mobile devices within an ad hoc wireless networks are assumed to have a limited computational power and scarce battery resources. This characteristics is even more true for sensor networks. Sensor networks are static networks built around the ad hoc networks' paradigm. Their goal is to make measurements (temperature, humidity, movement etc.) and and forward this data to a central point. A misbehavior detection mechanism for such networks must be therefore distributed, lightweight and adaptive. Many current misbehavior (intrusion) detection systems are nowadays designed as signature based systems that require that the set of misbehavior signatures gets updated often. This is clearly hardly possible for ad hoc networks. Is is also clear that an AIS might not provide the same level of protection as a human managed signature based system. Therefore it is of high importance to define which performability and structural properties of ad hoc networks should be subject to protection. We propose that an AIS for ad hoc networks should impose a high degree of their survivability [49]. It is therefore of paramount importance that the ad hoc network's mission is clearly defined and achievable under normal operating conditions. In the rest of this section we review some AIS based approaches to misbehavior (anomaly) detection for (ad hoc) wireless networks.

In [22] the authors describe an artificial immune system that is robust against anomalies at the transport layer of the OSI protocol stack; only wired TCP networks are considered. Self is defined as normal pairwise TCP connections. Instead of mimicking the complex structure of human immune defense they collapse B-cells and T-cells into a single entity called "detector". Each detector is represented as a single bit string of 49 bits. Such detector is able by string matching to recognize whether a given pair of TCP connections is self or non-self. The pattern matching is based on r-contiguous bits. A process of negative selection is applied to detectors in order to make them mature, i.e. to make them able to detect non-self. They assume that non-self behavior is very rare therefore training detectors on a running system is not unreasonable. They also introduced different activation and threshold conditions that make their system robust against incomplete sets of self that are used during detectors' training. The learning phase does not only include negative selection but also co-stimulation and a mechanism for maturing a detector into memory detectors. Co-stimulation is a secondary signal meant to suppress autoimmune reactions. In [22] when a detector matches a string (possible anomaly), a co-stimulation from a human network administrator is needed in order to confirm this string to be non-self. The time window in which the administrator can act was set to 24 hours; if no reply is received then the detector is reset, in other case the detector enters the competition to become a memory detector.

Additionally, in [17] the authors discuss the role of senescence for immune systems. They note that due to space efficiency memory cells will have to be eliminated over time. They also re-introduce the notion "ball of stimulation" that is based on research in the area of theoretical biology. Ball of stimulation models the fact that B or T-cells should be able to recognize non-self within the radius of the exact match. They also deal with holes caused by fixed-probability matching rules. They propose *permutation masks* associated with detector sets; a permutation mask controls how an antigen is presented to the detection system.

Ref. [34] discusses a network intrusion system that aims at detecting misbehavior by capturing TCP packet headers. They chose a more complex representation that accounts for traffic intensity, port used in the communication, TCP 3-way handshake unregularities, and ports are additionally tagged with known vulnerabilities if such exist. They report that AIS may be unsuitable for detecting anomalies in communication networks. This result is questioned in [6] where it is stated that the above negative result may be due to the choice of problem representation and to the choice of matching threshold r for r-contiguous bit matching. A positive result is also reported in [53] where several protocols from the network and transport layers are considered.

An interesting approach for detecting misbehavior is introduced in [46]. This approach builds on results in [22] and extends them in the direction of an artificial immune system

for detection of misbehavior at the network level of the OSI stack. The protocol that is subject to monitoring is DSR, or Dynamic Source Routing originally proposed by David Johnson et al.; see ref. [31]. The paper investigates the use of several novel concepts which are "virtual thymus", clustering for decreased rate of false positives, and a specific kind of costimulation called "danger signal". An approach for a more efficient secondary response is introduced as well.

The Danger theory by Aickelin et al. [3] suggests that recognition of a possible non-self is important only if this non-self is a relative danger to the system. This effort admits that the classical task of self–non-self recognition for an AIS is not sufficient and might even be unachievable due to non-efficiency of negative selection algorithms. It is however questionable how one can recognize a danger in ad hoc wireless networks as many performability measures require a global view of the network. In [46] the authors suggest that a source node should emit a danger signal when sent packets do not get acknowledged by the destination node. The signal is sent over the route to the destination. The signal contains information about the time when the packet was sent and about nodes that were supposed to forward this packet. This signal is then correlated with an observed non-self behavior (packet loss in this case). The authors do not discuss whether such a danger signal could self get misused.

In [13] we proposed that an AIS for ad hoc networks should consist of the following modules: Data collection and preprocessing, Local and cooperative detection, Learning, and Local and Cooperative response. These four layers should be mutually interconnected to allow for an efficient feedback mechanisms. This structure acknowledges the fact that for increased survivability of ad hoc networks is misbehavior detection equally important as finding the misbehaving node (or nodes), exchange of information about misbehaving nodes, and a possible cooperative corrective action against such nodes. A deficiency of current AIS for (ad hoc) wireless networks is that they concentrate on local detection of misbehavior; they do not consider distributed detection and with the exception of ref. [46] they only distantly cope with the problem of either local or cooperative response against a misbehaving node.

As suggested in [3] misbehavior detection and prevention should react on worsening performability and structural measures within an ad hoc network. It is obvious that in order to compute majority of these measures and thus to be able to determine the impact that a given misbehavior could have, it is necessary to have a global view of the network. In [7] we discuss what impact certain structural properties could have on performance of ad hoc networks. We conclude that when structural properties of an ad hoc network are known, the correlation between them and performability measures, such as throughput, latency or number of packets lost, is not clear. Therefore, motivated by results in [46], it seems one will have to limit itself to individual packet flows or to consider only performability measures that are easy to compute locally.

The above gives an outline of recent approaches that are applicable to wireless networks with artificial immunity. We expect that many of the approaches that were applied for wireless networks will be also applicable for ad hoc network. Since the field of AIS is still being in the early stages of development there is subsequently only very limited number of references that would deal with AIS for ad hoc networks. AIS and their usability for applications other than wireless networks are neatly reviewed in [4].

CONCLUSIONS

We have reviewed a specific area of anomaly detection systems. Artificial immune systems are based on properties of the Human immune system such as such as self vs non-self recognition, innate vs acquired immunity, primary vs secondary response, general vs specific response, or cell-mediated vs humoral immunity.

The key question of an AIS design is which structural and performability properties of the given (ad hoc) wireless network should be preserved. These invariants include connectivity and other graph theoretic measures [7], and a multitude of various performability parameters examples of which are packet latency, throughput, number of packets received or fairness.

We adhere to the idea that the an architecture for ad hoc wireless networks should impose a high degree of their survivability [49]. It is therefore desirable that the ad hoc network's mission is clearly defined and achievable under normal operating conditions.

Finally, we would like to point out that an AIS should never be expected to suppress an excessively large set of misbehavior. Therefore, when testing and training such a system the capability of misbehaving nodes should be clearly defined. On the other hand, any AIS system should be designed with some level of universality in mind, that is it should go beyond the current approaches that aim at protecting ad hoc networks against a specific flavor of misbehavior.

ACKNOWLEDGMENTS

This work was supported by the German Research Foundation (DFG) under the grant no. SZ 51/24-1 (Survivable Ad Hoc Networks – SANE).

REFERENCES

[1] ArachNIDS; advanced reference archive of current heuristics for network intrusion detection systems. `www.whitehats.com/ids`

[2] Imad Aad, Jean-Pierre Hubaux, Edward W. Knightly. Denial of service resilience in ad hoc networks. *Proc. 10th annual international conference on Mobile computing and networking*, 2004.

[3] U. Aickelin, P. Bentley, S. Cayzer, J. Kim, and J. McLeod. Danger theory: The link between ais and ids. *Proc. International Conference on Artificial Immune Systems (ICARIS'03)*, pages 156–167, Edinburgh, UK, 2003.

[4] Uwe Aickelin, Julie Greensmith and Jamie Twycross. Immune System Approaches to Intrusion Detection - A Review. *Proc. the 3rd International Conference on Artificial Immune Systems (ICARIS 2004)*, Catania, Italy, 2004.

[5] Tim Baas. Intrusion detection systems and multisensor data fusion. *Communications of the ACM*, vol. 43, issue 4, pp. 99-105, 2000.

[6] J. Balthrop and S. Forrest and M. Glickman. Revisiting lisys: Parameters and normal behavior. *Proc. Congress on Evolutionary Computing (CEC02)*, 2002.

[7] C. L. Barrett, M. Drozda, D. C. Engelhart, V. S. Anil Kumar, M. V. Marathe, M. M. Morin, S. S. Ravi, and J. P. Smith. Understanding Protocol Performance and Robustness of Ad Hoc Networks Through Structural Analysis. *Proc. IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob 2005)*.

[8] Mario Cagalj, Saurabh Ganeriwal, Imad Aad and Jean-Pierre Hubaux. On Cheating in CSMA/CA Ad Hoc Networks. Technical report No. IC/2004/27, February 2004.

[9] Alvaro A. Cardenas, Svetlana Radosavac, John S. Baras. Detection and prevention of MAC layer misbehavior in ad hoc networks. *Proc. 2nd ACM workshop on Security of ad hoc and sensor networks*, 2004.

[10] D'haeseleer, P. An immunological approach to change detection: theoretical results. *Proc. 9th IEEE Computer Security Foundations Workshop*, pp. 18 - 26, 1996.

[11] D'haeseleer, P., Forrest, S., and Helman, P. An immunological approach to change detection: Algorithms, analysis and implications. *Proc. IEEE Symposium on Research in Security and Privacy*, 1996.

[12] J. Douceur. The sybil attack. *Proc. of the IPTPS02 Workshop*, Cambridge, MA (USA), March 2002.

[13] M. Drozda, H. Szczerbicka, T. Bessey, M. Becker, R. Barton. Approaching Ad Hoc Wireless Networks with Autonomic Computing: A Misbehavior Perspective. *Proc. 2005 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS'05)*.

[14] Robert Durst, Terrence Champion, Brian Witten, Eric Miller, Luigi Spagnuolo. Testing and evaluating computer intrusion detection systems. *Communications of the ACM*, vol. 42, issue 7, pp. 53-61, 1999.

[15] Esponda, F., Forrest, S. and Helman, P. A formal framework for positive and negative detection, *IEEE Trans. Syst., Man Cybernet.*, vol. 34, pp. 357–373, 2004.

[16] Fernando Esponda, Stephanie Forrest, Paul Helman. The Crossover Closure and Partial Match Detection. *Proc. ICARIS 2003*, pp. 249-260.

[17] S. Forrest and S.A. Hofmeyr. Immunology as information processing. *In Design Principles for the Immune System and Other Distributed Autonomous Systems*, edited by L.A. Segel and I. Cohen. Santa Fe Institute Studies in the Sciences of Complexity. New York: Oxford University Press (2001).

[18] Fabio A. González, Dipankar Dasgupta, Luis Fernando Niño. A Randomized Real-Valued Negative Selection Algorithm. *Proc. ICARIS 2003*, pp. 261-272.

[19] V. Gupta, S. Krishnamurthy, and M. Faloutsos. Denial of Service Attacks at the MAC Layer in Wireless Ad Hoc Networks. *Proc. of Milcom*, 2002.

[20] P. Harmer, P. Williams, G. H. Gnusch, and G. Lamont. An Artificial Immune System Architecture for Computer Security Applications. *IEEE Transactions on Evolutionary Computation*, 6(3):252–280, June 2002.

[21] Emma Hart. Not All Balls Are Round: An Investigation of Alternative Recognition-Region Shapes. *Proc. ICARIS 2005*, pp. 29-42.

[22] S. Hofmeyr and S. Forrest. Immunity by Design: An Artificial Immune System. *Proc. Genetic and Evolutionary Computation Conference (GECCO)*, Morgan-Kaufmann, San Francisco, CA, pp. 1289-1296 (1999).

[23] Yih-Chun Hu, Adrian Perrig, David B. Johnson. Ariadne:: a secure on-demand routing protocol for ad hoc networks. *Proc. 8th annual international conference on Mobile computing and networking*, 2002.

[24] Yih-Chun Hu, Adrian Perrig, David B. Johnson. Rushing attacks and defense in wireless ad hoc network routing protocols. *Proc. ACM workshop on Wireless security*, 2003.

[25] Yih-Chun Hu, Adrian Perrig, David B. Johnson. Packet leashes: a defense against wormhole attacks in wireless networks. *Proc. INFOCOM 2003, Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies*.

[26] Sami Iren, Paul D. Amer, Phillip, T. Conrad, The transport layer: tutorial and survey, *ACM Computing Surveys*, vol. 31, no. 4, pp. 360-404, 1999.

[27] M. Jakobsson and S. Wetzel and B. Yener. Stealth attacks on ad hoc wireless networks. *Proc. VTC*, 2003.

[28] Charles A. Janeway Jr. How the immune system works to protect the host from infection: a personal view. *Proc. Natl. Acad. Sci. U S A.*, 2001 Jun 19;98(13):7461-8.

[29] N.K. Jerne. Towards a network theory of the immune system. *Annuals of Immunology*, 1974.

[30] Zhou Ji, Dipankar Dasgupta. Real-Valued Negative Selection Algorithm with Variable-Sized Detectors. *Proc. Proc. Genetic and Evolutionary Computation Conference 2004 (GECCO-2004)*, 2004: 287-298

[31] D. Johnson and D. Maltz. Dynamic Source Routing in Ad Hoc Wireless Networks. *Mobile Computing*, Tomasz Imielinski and Hank Korth, Eds. Chapter 5, pages 153-181, Kluwer Academic Publishers, 1996.

[32] A. Juels and J. Brainard. Client puzzles: A cryptographic countermeasure against connection depletion attacks. *In Distributed Systems Security (SNDSS)*, pages 151–165, February 1999.

[33] Jeffrey O. Kephart, David M. Chess. The Vision of Autonomic Computing. *IEEE Computer magazine*, January 2003.

[34] Kim, J. and Bentley, P. J. Evaluating Negative Selection in an Artificial Immune System for Network Intrusion Detection, *Proc. Genetic and Evolutionary Computation Conference 2001 (GECCO-2001)*, San Francisco, pp.1330 - 1337, July 7-11, 2001.

[35] P. Kyasanur and N. H. Vaidya. Detection and handling of mac layer misbehavior in wireless networks. Technical report, CSL, UIUC, August 2002.

[36] Howard F. Lipson and David A. Fisher. Survivability - A New Technical and Business Perspective on Security. *Proc. 1999 New Security Paradigms Workshop*, Association for Computer Machinery, 2000.

[37] Sergio Marti and T. J. Giuli and Kevin Lai and Mary Baker. Mitigating routing misbehavior in mobile ad hoc networks. *Mobile Computing and Networking*, pp. 255-265, 2000.

[38] Biswanath L. Mukherjee, Todd Heberlein, and Karl N. Levitt, Network Intrusion Detection. *IEEE Network*, vol. 8 no. 3, pp. 26-41, May/June 1994.

[39] Steven Noel, Duminda Wijesekera, Charles Youman. Modern Intrusion Detection, Data Mining, and Degrees of Attack Guilt. *In Applications of Data Mining in Computer Security*, D. Barbarà and S. Jajodia (eds.), Kluwer Academic Publisher, 2002.

[40] Venkata N. Padmanabhan, Daniel R. Simon. Secure traceroute to detect faulty or malicious routing. *ACM SIGCOMM Computer Communication Review*, vol. 33, issue 1, pp. 77-82, 2003.

[41] Charles E. Perkins. *Ad Hoc Networking*. Addison Wesley, 2001.

[42] Adrian Perrig, John Stankovic, David Wagner. Security in wireless sensor networks. *Communications of the ACM*, vol. 47, issue 6, pp. 53-57, 2004.

[43] T.S. Rappaport. *Wireless Communications*. Prentice-Hall, 1996.

[44] Stefan Savage, Neal Cardwell and David Wetherall and Tom Anderson. TCP Congestion Control with a Misbehaving Receiver. *Computer Communication Review*, vol. 29, number 5, 1999.

[45] Stefan Savage, David Wetherall, Anna Karlin, Tom Anderson. Network support for IP traceback. *IEEE/ACM Transactions on Networking*, vol. 9, issue 3, pp. 226-237, 2001.

[46] Slaviša Sarafijanović and Jean-Yves Le Boudec. An Artificial Immune System for Misbehavior Detection in Mobile Ad-Hoc Networks with Virtual Thymus, Clustering, Danger Signal and Memory Detectors. *Proc. ICARIS (Third international conference on artificial immune systems)*, 2004.

[47] S. Singh. Anomaly detection using negative selection based on the r-contiguous matching rule. *Proc. 1st International Conference on Artificial Immune Systems(ICARIS)*, pp. 99–106, 2002.

[48] Frank Stajano, Ross Anderson. The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks. Security Protocols, *Proc. 7th International Workshop*, 1999.

[49] James P. G. Sterbenz, Rajesh Krishnan, Regina Rosales Hain, Alden W. Jackson, David Levin, Ram Ramanathan, John Zao. Survivable mobile wireless networks: issues, challenges, and research directions. *Proc. ACM workshop on Wireless security*, 2002.

[50] Stibor, T., J. Timmis und C. Eckert. A Comparative Study of Real-Valued Negative Selection to Statistical Anomaly Detection Tech niques. *Proc. 4th International Conference on Artificial Immune Systems (ICARIS-2005)*, Banff, Canada, 2005.

[51] Bo Sun, Kui Wu, Udo W. Pooch. Alert aggregation in mobile ad hoc networks. *Proc. ACM workshop on Wireless security*, 2003.

[52] Giovanni Vigna, Fredrik Valeur, Richard A. Kemmerer. Designing and implementing a family of intrusion detection systems. *Proc. 9th European software engineering conference*, 2003.

[53] Paul D. Williams, Kevin P. Anchor, John L. Bebo, Gregg H. Gunsch, Gary D. Lamont. CDIS: Towards a Computer Immune System for Detecting Network Intrusions. *Proc. RAID 2001*, LNCS 2212, pp. 117-133, Jan 2001.

[54] Hao Yang, Xiaoqiao Meng, Songwu Lu. Self-organized network-layer security in mobile ad hoc networks. *Proc. ACM workshop on Wireless security*, 2002.

[55] Yongguang Zhang, Wenke Lee, and Yian Huang. Intrusion Detection Techniques for Mobile Wireless Networks, *ACM/Kluwer Wireless Networks Journal (ACM WINET)*, vol. 9, No. 5 (September 2003).

[56] Lidong Zhou and Zygmunt J. Haas. Securing Ad Hoc Networks. *IEEE Network*, vol. 13, number 6, pp. 24-30, 1999.