Approaching Ad Hoc Wireless Networks with Autonomic Computing: A Misbehavior Perspective

Martin Drozda, Helena Szczerbicka, Thomas Bessey, Matthias Becker, Rainer Barton
University of Hannover, Department of Computer Science,
FG Simulation und Modellierung, Welfengarten 1, 30167 Hannover, Germany.
Email: {drozda,hsz,tby,xmb}@sim.uni-hannover.de

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

Abstract

We review recent approaches to dealing with misbehavior in ad hoc wireless networks. We focus on a specific class of solutions that are based on autonomic computing. This class is motivated by the very efficient and complex system that is able to protect the health of humans against an amazing set of malicious extraneous attacks. We also provide the reader with a summary of misbehavior that is currently considered in the literature. Certain aspects of machine learning and game theory with relevance to misbehavior detection are reviewed as well. Based on relevant design properties of Artificial immune systems motivated by human immunity we present an outline of a four-layer architecture for ad hoc wireless networks. The purpose of this architecture is to impose a high degree of survivability against misbehavior of nodes.

MOTIVATION

With the increasing complexity of wireless networks, the task of network management is becoming more difficult to handle. This is a consequence of their heterogeneity and their objectives set. One of the objectives that future wireless networks will need to fulfill is to stay robust or immune against external intruders that attempt to disrupt or interrupt their operations. In a position paper on autonomic computing [36], the authors point out that *soon systems will become too massive and complex for even the most skilled system integrators to install, configure, optimize, maintain, and merge.* They suggest that the answer to the phenomenon are systems that would be fully autonomic - systems that would be able to manage themselves following only high level objectives from system administrators.

The objective of this review are approaches to self protective systems. We derive from it a design of a system for ad hoc wireless networks that would be fully autonomous in detecting anomalous behavior in such networks, a system that would be capable of learning about anomalies, and capable of disseminating this knowledge throughout the network, possibly able to isolate the source of misbehavior. Such a system has the property of survivability, that means a capability of preserving services at an acceptable level after a disturbance in a system. These ideas come along the successful efforts in the area of Artificial Immune Systems (AIS) that are based on mechanism that are present in human bodies, namely, on *human immunity system (HIS)*; see [17, 21, 56, 67, 3] and references therein. We are also motivated by recent advances in Intrusion detection systems [47, 65, 63, 50, 14, 5, 60, 68, 45].

INTRODUCTION

The paradigm of wireless ad hoc networks[1] is *connectivity anywhere, at any time, without any fixed infrastructure*. Pioneering work in this area has been done in the SURAN [57] and PRNET [34] project sponsored by the DARPA agency at the US Department of Defense. Nowadays, this paradigm has been embraced by major commercial and military contractors worldwide.

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. Ability to keep network connected is one of important characterizations of ad-hoc network. A 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 user. 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.

We would like to add that the battery power that is necessary at each node for reception or transmission of data packets, and for all necessary computation as prescribed by

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

different protocols is of rare nature and therefore has to be preserved. 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 [49, 52, 27]. 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. The standard index for long term fairness is Jain's Index [28]. Other QoS parameters include overhead at different layers of the OSI stack[2], spatial use of control packets [6], 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. In the next few sections we review literature that discusses i) flavors of misbehavior in wireless (ad hoc) networks, ii) intrusion and misbehavior detection techniques, iii) basics of human immunity with affinity to survivability, iv) state-of-the-art in design of artificial immune systems and as well as a few other related issues.

MISBEHAVIOR AND APPROACHES FOR ITS MITIGATION

Misbehavior in Wireless (Ad hoc) Networks

In this section we review a few known types of misbehavior that can lead to decreased Quality of Service in wireless networks. 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][3]; 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 physical layer is not of importance.

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 [11, 38, 10, 20].

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 [38].

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 [30].

Manipulation of routing tables, route caches, and data structures with routing information. An attack aimed at originating inconsistencies in network and creating collisions; see [55, 65]. Ref. [62] discusses an interesting manipulation by creating bogus RREP packets. Ref. [48] 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.

Packet forwarding misbehavior is usually understood as packet dropping, packet duplicating, and packet jamming. It can be partially eliminated by the Watchdog technique [41]; the assumptions are that the given hardware device is able to function in promiscuous mode and that power level control

[2]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.

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

and directional antennas are not used.

Impersonation or IP spoofing is performed by introducing packets that have stated originators different from real; see [55].

Sybil attack is done by creating a number of fictitious nodes; see [13].

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 [35].

ACK division, DupACK spoofing, and optimistic ACKing. This misbehavior is aimed at manipulation of the size of the congestion window at senders; see [54].

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.

Human Immune System

The Human immune system (HIS) 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 detecting foreign antigens. An antigen is anything that can initiate an immune response; examples of antigens are a virus, bacteria, or splinter.

The ability of HISs to protect humans against antigens is due to certain features that are a result of evolution, the reasons for the existence of these features are often very complex and beyond the scope of this review. Often certain phenomena are explained only through hypotheses, an example of a popular hypothesis is the theory of idiotypic networks by Jerne [32].

The important features of HISs have often a dual nature. These dual natures include self vs non-self recognition, innate vs acquired immunity, primary vs secondary response, general vs specific response, or cell-mediated vs humoral immunity. Certain immunity mechanisms are antigen specific (they act only against specific antigens), 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).

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.

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 replicate 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. Memory B-cells are created. 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.

B-cells mature in bone marrow, T-cells mature in thymus. A B-cell is stimulated to maturity when an antigen binds to its surface receptors. This causes a binding by a helper T-cell and the B-cell matures. A T-cell is coated with various substances or clusters of differentiation. These clusters of differentiation allow for a large variability in in the form of receptors. T-cells are primed in thymus. First they face the process of positive selections. At this stage, T-cells that happen to recognize any self cell are destroyed. At the next stage, only T-cells that happen to recognize an antigen are allowed to pass out; this process is called negative selection.

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. Acquired immunity is a result of a complex learning system.

For details on human immunology we refer the interested reader to classical texts such as [31]. 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.

We would like to stress that the objective of any misbehavior detection AIS should not be to mimic the mechanism of

HISs, but rather to motivate design by them. The architecture of human immunity is heavily dependent on the chemical and biological processes within HISs. In [21], the authors state that the most important issues motivated by HISs that should be incorporated into any AIS should be: self/non-self recognition, robustness, distributed nature, error tolerance, adaptivity, and dynamics. Mapping human immunology to computable states and procedures is not a straightforward task. In the following section we will review efforts in this direction, point out their weaknesses and discuss their feasibility.

Mapping Immunity to Computation

Pioneering work in the direction of mapping immunity to networking has been done by S. Forrest and her group at University of New Mexico. In their position papers [17, 21] they stress that only the most relevant features of HISs should be incorporated in the construction of artificial immune systems.

In [21] they 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"[4]. 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, that is 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 mechanism for maturing a detector into memory detectors.

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 lymphocyte (B or T-cell) should be able to recognize antigen within the radius of the exact match. It is obvious that such balls of stimulation can be implemented by Hamming distance, or bit by bit comparison with threshold as it was done in [21].

An interesting approach for detecting misbehavior is introduced in [56]. This approach builds on results in [21] 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. [33]. 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 co-stimulation called "danger signal". An approach for a more efficient secondary response is introduced as well.

The above gives an outline of recent approaches that are *strictly* applicable to ad hoc networks with artificial immunity. We are aware of other efforts; these efforts are neatly reviewed in [3].

The drawbacks of the above discussed approaches can be summarized in the following:

- Adversary modeling is missing; it is often not clear what misbehavior is expected from an intruder, what damages it can produces. A formal attempt in this direction is present only in [4].
- Time and space consideration for node computations are not discussed. It is often assumed that memory space at nodes is unlimited. Fusion of antigen information is not considered in order to preserve space.
- Matching of detectors and antigens is done at a rather simplified level which is usually string matching or a similar technique. Approaches based on similarity hashing, similarity searching, clustering or other techniques that have found application in the areas of e.g. genome databases are not considered.
- It is often not clear whether the new architecture for artificial immunity does not introduce new security holes. If nodes can misbehave in routing, why cannot they misbehave in forwarding a danger signal that is for example implemented in [18]?
- Mapping between human immunology and artificial immunology is frequently not well described and motivated. Issues such as mapping an ad hoc network to a human body, or mapping nodes to human bodies are discussed only vaguely. Ref. [3] points out that at this stage of research maturity, it is not clear whether packets or streams of packets should be tested against self and non-self.
- Frequently it is not obvious what are the objectives of the artificial immune system, what invariants and performability parameters should be preserved, and which can be surrendered.
- Mapping is often done on a single layer of the OSI stack. Researchers either consider a routing, transport or other protocol; furthermore they do so in isolation.
- Recent trends in ad hoc networking such as integration of protocols at different layers of the OSI stack [6, 53, 39] are not considered.

It is of general understanding that due to the infrastructureless nature of ad hoc networks, their robustness

[4] In the following text will be "detector" and "detector cell" used interchangeably.

against various kinds of misbehavior or anomalies is more important. According to [9], security attacks can be divided into two basic groups: a) attacks on a basic mechanism where under a basic mechanism it is understood the underlying protocols and b) attacks on security mechanisms such as key management. In this review we only deal with the first group of security issues.

Unsupervised Learning in Ad Hoc Networks

As we already mentioned, self vs non-self recognition is of a great importance for any AIS motivated by human immunity. We assume that this recognition is done autonomously at each node; that means each node has to have means for discriminating self and non-self behavior, storing and manipulation self and non-self patterns efficiently, aging of these patterns (senescence) and exchanging self and non-self information among all nodes participating in an ad hoc network.

A useful approach to identification of self and non-self information is clustering. Clustering, a sub-class of unsupervised learning, is a way to form a natural grouping of patterns. It makes a system to learn to represent particular input patterns in a way that reflects the statistical structure of the overall collection of input parameters. By contrast with supervised or reinforcement learning there is no explicit target output associated with each input.

Clustering algorithms classify a set of data into groups, so that similar data is grouped together. In our case packet streams could be observed and those with similar characteristics would be included into the same class. If a packet stream is grouped together with other packet streams which represent a tolerable communication pattern, then we would suppose that the new packet stream should be accepted as 'self'. If a packet stream cannot be classified as 'self' then a different form of classification has to be utilized. A straightforward solution is a co-stimulatory mechanism such as danger signal [56]. Danger signal can be connected to important QoS parameters. An example would be a new packet stream that is connected e.g. with an unusual increase in packet latency or with anomalous changes in routing tables. A sudden change in any important QoS parameter is non-arguably a reason for action.

Cluster analysis is the organization of a collection of patterns (usually represented as a vector of measurements, or a point in a multidimensional space - here the data and control content of packets) into clusters based on similarity. Intuitively, patterns within a valid cluster are more similar to each other than they are to a pattern belonging to a different cluster.

The quality and the amount of data is a necessary condition for a success in assessing its true class structure. The problem in clustering is to group a given collection of unlabeled patterns into meaningful clusters. In a sense, labels are associated with clusters also, but these category labels are data driven, that is, they are obtained solely from the data. Typical pattern clustering activity involves pattern representation, definition of a pattern proximity, clustering, data abstraction and assessment of output.

The most popular metric for similarity in unsupervised algorithms is the Euclidean distance, but there are some other distance measures reported in the literature that take into account the effect of surrounding or neighboring points. These surrounding points are called context in [42]. A "metric" using context is the mutual neighbor distance (MND) proposed in [19]. The MND is not a metric in a strict way, it does not satisfy the triangle inequality. In spite of this fact, MND has been successfully applied in several clustering applications. This observations supports the viewpoint that dissimilarity does not need to be a metric.

Two classes of approaches have been suggested to clustering data. Density estimation techniques explicitly build statistical models (such as Bayesian networks) of how underlying causes could create the input. Feature extraction techniques try to extract statistical regularities (or irregularities) directly from inputs.

There is a distinction between hierarchical and partitional approaches (hierarchical methods produce a nested series of partitions, while partitional methods produce only one). Most hierarchical clustering algorithms are variants of the single-link [58], complete-link [37] and minimum-variance [64, 46] algorithms. In the single link method, the distance between two clusters is the minimum of the distances between all pairs of pattern drawn from the two clusters. In the complete link algorithm the distance between two clusters is the maximum of all pairwise distances between patterns in the two clusters. In either case, two clusters are merged to form a larger cluster based on minimum distance criteria. The clusters obtained by the complete link algorithm are more compact than those obtained by the single link algorithm. The single link algorithm is, however, more versatile than the complete link algorithm.

It has been observed [29] that the complete link algorithm produces more useful hierarchies in many applications than the single link algorithm. Hierarchical algorithms are more versatile than partitional algorithms. On the other hand, the time and space complexities of the partitional algorithms are typically lower than those of the hierarchical algorithms [12].

Ref. [66] developed the concept of a two-tier intrusion detection system based on clustering with Kohonen's Self Organizing Maps. Their work hypothesis is that on the most networks, the traffic would belong to a small number of services and protocols that are regularly used, and so that most of it would belong to a relatively small number of classes. In the first tier of their system, an unsupervised clustering algorithm classifies the payload of the packets, observing one packet at a time and compressing it into a single byte of information. This classification is added to the information decoded from the packet header and passed on to the second tier. The second tier algorithm instead takes into consideration the anomalies, both in each single packet and in a sequence of packets. This

approach shows promising results.

Adversary Modeling with Game Theory

The abilities of adversaries have to be well defined in order to facilitate a successful design of an AIS. It is unreasonable to expect that an AIS would be able to shield an ad hoc network from an arbitrary type of misbehavior. Game theory [22] offers a suitable definition of two powerful classes of misbehaving nodes.

Under this theory, two basic types of nodes participating in an ad hoc network are, namely selfish and malicious nodes. The latter nodes attack the network in some way in order to disturb its normal operation; such ways of attacking may include network flooding, manipulation of forwarded packets or simply the denial of packet forwarding. Selfish nodes are different in that they always act just for their convenience while not interested in harming the network. While selfish nodes may decide not to forward packets just like malicious nodes do, they just do so in order to e.g. save energy for their own communications (as opposed to malicious nodes). Of course, the effect of denial of packet forwarding is the same in both cases, namely, the connectivity of the ad hoc network (and with this, any related performance measure such as throughput) is likely to suffer. However, selfish nodes will experience this effect having an impact on their own connectivity/performance; consequently, they will try to avoid it since they act for their very own convenience. In other words, while there is not any chance to turn malicious nodes into non-malicious nodes, selfish nodes are willing to act "fairly" in terms of the network if this is in their own interest.

We assume that misbehavior will be mostly exercised in collusion and, to a lesser extent, by single nodes; see [68, 23, 24, 30, 55, 4, 8, 10, 62, 67, 38, 51] for references on node misbehavior modeling.[5] Conclusions that can be drawn from examples in these references are: i) nodes have freedom to disobey the mechanism prescribed by protocols, ii) nodes are not expected to be tamper proof over a long term horizon, iii) program and user activities are observable, iv) functionality updates are expensive and subject to attacks, v) all nodes can be expected to exhibit selfish behavior.

Several groups of researchers have proposed different incentives mechanisms, in order to foster cooperation between nodes [8, 44]. The need for an incentives approach is frequently not formally justified; we refer the reader to [26] for an interesting discussion of this approach. This motivates the part of the literature on ad hoc networks which proposes application of game theory in order to study selfish node behavior and to analyze the resulting performance of networks.

Game theory is a mathematical discipline founded as a strict theory in the fifties by John von Neumann and Oscar Morgenstern. This theory provides a formal framework for the modeling and analysis of scenarios where at least two different parties with own interests, called (selfish or rational) "players", deal with each other about some outcome; this process is called a "game". The desired outcome is given by the players' own interests, while the actual outcome is the result of their mutual responses to the actions of their counterparts in order to reach the desired outcome. (The actions of players may not be observable to other players; in such case, players have to make reasonable assumptions about the other players' actions.) A player always takes appropriate actions in order to reach his desired outcome; the sequence of actions taken is called his "strategy". The evaluation of actions in terms of the outcome is done by means of the so-called "utility function".

In ad hoc networks, selfish nodes (players) typically do not cooperate, i.e., they do not mutually negotiate strategies in order to keep the network operating as close to the optimum as possible, because such agreement simply cannot be ensured to be honored since there is no authority to do so. Consequently, the interaction of nodes in an ad hoc network is modeled by a non-cooperative game. Generally, a single player of a non-cooperative game tends to deviate from a "loose" agreement (without supervision by an authority) as this typically will improve his own benefits. However, in many non-cooperative games, there is at least one combination of strategies for all players where no player has an incentive to deviate from his strategy unilaterally since this will not improve his outcome. Such combination is called a "Nash equilibrium". However, a Nash equilibrium generally does not result in an optimal (so-called "Pareto optimal") outcome of the game, e.g., maximal throughput of an ad hoc network. Any application of game theory in ad hoc networks involves studying cooperation in such networks by identifying Nash equilibria.

In [15], the authors focus on the data link layer in that they study topology control problems, where network nodes get to choose their power levels in order to ensure desired connectivity properties. Each node is thought of as a player; in the games they consider, a player needs to choose a radius, and a choice of the radius is a strategy. In the considered game about connectivity, each node aims to minimize its radius. The authors then study Nash equilibria and show that (among the games they define) these can only be guaranteed to exist if all network nodes are required to be connected to all other nodes.

In [59], the authors assume that each node is associated with a minimum lifetime constraint. Again, each node is thought of as a player. Given the lifetime constraints and the assumption of rational behavior, the authors are able to determine the optimal throughput that each node should receive. Basically, for each node, the ratio of the number of successful relay requests generated by the node, to the number of relay requests made by the node is used as an indication of the throughput experienced by the node. The authors then propose a distributed and scalable acceptance algorithm based on a well-known strategy in game theory called "Tit-

[5] We expect that certain simpler types of misbehavior can be already mitigated by current mechanisms in protocols; an example would be an RERR packet of a routing protocol notifying other nodes about link failure.

For-Tat", where basically, nodes react on other nodes' actions exactly in the same way; the acceptance algorithm is used by the nodes to decide whether to accept or reject a relay request. The scalability of the algorithm is achieved by assuming some energy classes for all nodes; with this, a node does not maintain individual records of its experience with every node in the network since the interaction between nodes is dominated by the node with the smallest power constraint. The authors show that the algorithm results in a Nash equilibrium and prove that the system converges to a Pareto optimal operating point. However, calculation of the optimal ratios as introduced above requires each user in the system to be aware of the number of users in each energy class and the energy constraint for each class. The authors state that they need to devise a distributed mechanism to acquire and disseminate the necessary information to all users. They also state that the mechanism should be sufficiently robust to prevent (malicious) users from propagating incorrect information to serve their own needs.

In [16], the authors prove several theorems about the equilibrium conditions in a simple scenario. They model packet forwarding as a game where each node as a player interacts with the rest of the network without identifying the players it interacts with. The scenario under study is very simple in order to make analysis by means of game theory possible; basically, the authors assume that the nodes are organized in a ring, where each connection has one relay. Each node decides for each packet whether to forward it or not, using its own strategy. The strategy of a node is defined by means of the ratio of the number of packets that were originated at the node and were successfully received at the destinations (benefit of the node), to the number of packets that the node forwarded for other nodes (contribution of the node). The node forwards packets only if its ratio exceeds some threshold. The authors then investigate by simulation a more realistic scenario, which includes a real network topology as well as a mobility model. They show that the level of contribution of the nodes to reach cooperation is much higher than in the theoretical model, and they quantify the relationship between mobility and cooperation.

In [43] have the authors proposed a generic mechanism ("security scheme") based on reputation to enforce cooperation among the nodes of an ad hoc network and to prevent passive denial of service attacks due to node selfishness (e.g., denial of packet forwarding). In the cited work, they propose a game theoretical approach in order to analyze the robustness of the mechanism. They show that, while nodes of an ad hoc network where no security scheme is adopted will eventually free ride, the best strategy a node could choose in the presence of their mechanism is to collaborate. More specifically, the authors show that, under certain conditions, the mechanism assures the cooperation of at least half of the nodes of an ad hoc network.

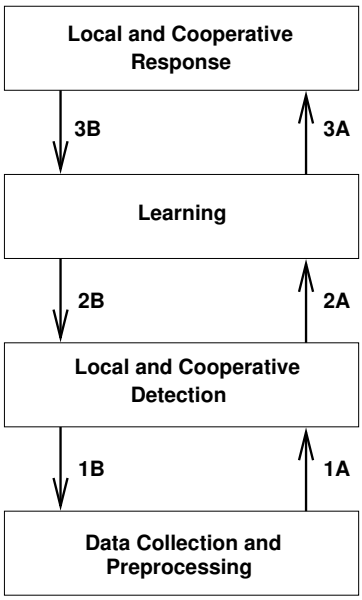

Figure 1. Architecture of our artificial immune system. It consists of four layers that are signaling and feedback enabled.

AN ARCHITECTURE FOR SURVIVABLE AD HOC NETWORKS

General Design Guidelines

Our general design guideline is an architecture for wireless ad hoc networks that would impose a high degree of 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* [40]. We define mission as the capability of keeping network throughput as high as possible. Our solution to survivable ad hoc networks is an architecture motivated by qualities of human immune systems. In contrast to existing approaches, which are tailored to specific scenarios, our approach aims at universality at a reasonable level being not restricted to particular protocols.

The Architecture

We propose a distributed four-layer architecture with the following modules: Data collection and preprocessing, Local and cooperative detection, Learning, and Local and Cooperative response; see Figure 1. Each module is expected to communicate directly with other module above or below. This is in order to guarantee information flow and feedback capability. Unlike in human immunity systems, immunity functions are not centralized in specific nodes but implemented in each node. We note that the core module Learning will implement several principles of learning as found in human immune systems such as presented in Section .

In the following we elaborate in more detail on the individual modules. A kernel based on this four layer architecture

will be an integral part of each node.

1. **Data Collection and Preprocessing:** Data at the three layers of the OSI stack is collected. The data that will be atomic events in a sliding window (with adaptive size) are for example updates to routing tables, transmission of control packets by any protocol, forwarding of data packets etc. This data will be collected both in the node as well as around the node by working in promiscuous mode. The challenge of this module is to gather information that can be efficiently used for anomaly detection respecting restrictions of a node. These restrictions are imposed by limited battery power, limited computational capability, and limited memory. Therefore mechanisms for aging information, adaptive windowing, and time and space computational limits have to be considered. Such data is expected to show a high degree of multidimensionality.
2. **Local and Cooperative Detection:** Basic properties of a detection are: i) ability to detect any new behavior, ii) pattern acquisition of such a behavior. Local detection of anomalous behavior can be done on basis of unsupervised learning such as clustering. By clustering of data and its pattern acquisition we expect to be able to efficiently identify behavior that has not been observed in the past. Clustering has been already successfully applied to such problems; see [66]. The prerequisites for successful clustering in our context are: i) computational feasibility, ii) effective pattern extraction.

   Local detection can be extended to cooperative detection through exchanging of data patterns among participating nodes.
3. **Learning:** Learning can be done through negative selection, maturation of naive cells into memory cells, information proliferation, co-stimulation etc. Negative selection is done by removing cells that match any self behavior from the system. Their maturation is done by signaling (co-stimulation) from detection layer that this cell matches a newly observed behavior or by a danger signal. A danger signal is created when basic performability measures observable at the node statistically change. A danger signal can be also received from other nodes, however, at a cost of a possible attack on such a signal. Simple versions of danger signals were successfully used in scope of artificial immune systems, see e.g. [56].

   Learning can be extended to collective learning. This can be done by statistical analysis of self and non-self behavior over a set of nodes.
4. **Local and Cooperative Response:** A response is understood to be a series of actions that: i) eliminates or decreases the impact of misbehavior to the node, ii) decreases the rate of misbehavior proliferation over the networks, iii) identifies and/or isolates the source of misbehavior. In humans such a response is done through learning of successful remedies and their application. The cardinality of the set of such responses is usually fixed; some responses are artificially introduced through vaccination. Our vision is to design a module that can: i) learn which response is efficient, ii) launch a more powerful secondary response once that type of misbehavior has been already observed in the past.

Let us consider an ad hoc network consisting of a set of nodes that may behave selfishly or maliciously. Each node is auditing the event stream that he observes. This stream is a series of events such as RTS sent, CTS received, TCP DATA packet sent, CTS sent, etc. For each of these events, attributes such as from and to address field, delay between forwarding and TCP ACK received, delay between an RTS and and an CTS, etc. are recorded at the Data collections and preprocessing module. This observation is done over a sliding window with an adaptive size.

This data is formatted and preprocessed and subsequently sent to the Layer Local and Cooperative Detection (1A[6]) so that it can be subject to clustering or other classification method. When this is not possible, a feedback (1B) from the Detection layer is received and either the window size has to be changed, the data collected has to get split or reaggregated, or the data detail has to be increased. On preprocessed data classification in the form of clustering, pattern matching etc. can be done.

When a pattern for the data exists (data exchange 2A-2B necessary) and the behavior is known to be non-self, a secondary learned response can be launched (this is signaled by 3A). Otherwise, this behavior has to be observed with an over-threshold frequency, and classified.

Classified data is received by the Learning layer over 2A. It will be subject to negative selection (or other learning methods); this assures that the pattern for it is unique and not conflicting with any self behavior pattern. A detector cell that survives negative selections is made mature when an additional co-stimulation signal is received. This co-stimulation is created when a performance measure starts showing worsened characteristics, for example when the given node does not receive TCP ACKs for its sent data packets. External co-stimulation in form of a danger signal from other nodes is also possible. In this case, such behavior is classified as non self. It can be later reclassified or get completely removed.

A response from a set of known remedies is applied. Such a remedy can be for example a random packet dropping of packets from a chosen host or assigning them a lower priority. Exchange 3A-3B is necessary in order to evaluate response effectiveness.

[6] See Figure 1.

CONCLUSIONS

We have reviewed a specific area of misbehavior detection and mitigation. Procedures that we described are based on human immune systems and form a subcategory of Artificial immune systems. These AISs are based on well known properties of HISs 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. We pointed out that due to the distributed nature of ad hoc wireless networks it will be necessary to design and implement highly survivable systems that will be able to deal with the types of misbehavior that is often discussed in the literature. Moreover, as ad hoc networks become more popular it is reasonable to expect that the number and type of misbehavior will grow very fast with the number of users.

The key question of an AIS design is which structural and performability properties of the given ad hoc 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 wireless ad hoc networks should impose a high degree of their survivability [61]. 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* [40]. It is therefore desirable that the ad hoc network's mission is clearly defined and achievable under normal operating conditions.

We propose that an architecture for the next generation artificial immune systems that should be able to provide ad hoc networks with a certain degree of survivability has the following basic modules: Data collection and preprocessing, Local and cooperative detection, Learning, and Local and Cooperative response. These four layer are mutually interconnected to allow for efficient feedback mechanisms. Algorithms and data structures within these four layers should be focused on prevention and detection of intrusions and attacks rather than on undoing damage caused by them. It is questionable how techniques that maintain a certain degree of connectivity and a certain level of battery power will be able to help as they share the same manipulation and exploitation weaknesses as approaches based for example on attack signature propagation. However, we recognize the necessity for individual nodes to exchange misbehavior information and, if necessary, to act in collusion in order to identify and possibly neutralize sources of misbehavior.

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.

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] 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.

[4] Baruch Awerbuch, David Holmer, Herbert Rubens. "Provably Secure Competitive Routing against Proactive Byzantine Adversaries via Reinforcement Learning", preprint, 2003.

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

[6] Christopher L. Barrett, Martin Drozda, Achla Marathe, and Madhav V. Marathe. Characterizing the Interaction Between Routing and MAC Protocols in Ad-Hoc Networks. Proc. The Third ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC 2002), June 2002, Lausanne, Switzerland.

[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. IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob 2005). To appear.

[8] S. Buchegger and J. Le Boudec. Performance Analysis of the CONFIDANT Protocol: Cooperation Of Nodes — Fairness In Dynamic Ad-hoc NeTworks. Proceedings of IEEE/ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC), Lausanne, CH, 2002.

[9] Levente Buttyán, Jean-Pierre Hubaux. "Report on a working session on security in wireless ad hoc networks." ACM SIGMOBILE Mobile Computing and Communications Review, Volume 7 , Issue 1, pp. 74 - 94, 2003.

[10] 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.

[11] 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.

[12] Day, W.H.E. Complexity Theory: An Introduction for Practitioners of Classification. In Clustering and Classi-

fication, P. Arabie and L. Hubert, Eds, World Scientific Publishing Co., Inc., River Edge, NJ, 1992.

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

[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] S. Eidenbenz, V.S. Anil Kumar, S. Zust. Equilibria in Topology Control Games for Ad Hoc Networks. Los Alamos National Laboratory Report LA-UR-03-3269, 2003.

[16] M. Felegyhazi, J.-P. Hubaux, L. Buttyan. Equilibrium Analysis of Packet Forwarding Strategies in Wireless Ad Hoc Networks – the Dynamic Case. EPFL-IC Technical Report IC/2003/68, 2003.

[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] Aristides Gionis, Piotr Indyk, Rajeev Motwani. "Similarity Search in High Dimensions via Hashing." Proc. the 25th International Conference on Very Large Data Bases, pp. 518 - 529, 1999.

[19] Gowda, K. C. and Krishna, G. 1977, Agglomerative clustering using the concept of mutual nearest neighborhood, Pattern Regn. 10, pp. 105-112.

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

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

[22] M.J. Holler, G. Illing. Einführung in die Spieltheorie. Springer-Verlag; 3. Auflage, 1996.

[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. INFOCOM 2003, Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies.

[26] Elgan Huang, Jon Crowcroft, Ian Wassell. Rethinking incentives for mobile ad hoc networks. Proc. ACM SIGCOMM workshop on Practice and theory of incentives in networked systems, 2004.

[27] 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.

[28] R. Jain, "The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling,"Wiley- Interscience, New York, NY, April 1991.

[29] Jain and Dubes, Algorithms for Clustering Data, Prentice Hall Advanced Reference Series. Prentice Hall, Inc., Upper Saddle River, N.J., 1998.

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

[31] 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.

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

[33] 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.

[34] J. Jubin and J. D. Tornow. The DARPA Packet Radio Network Protocols. *Proceedings of the IEEE,* 75(1), pp. 21-32, January 1987.

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

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

[37] King, B. 1967, Step-wise clustering procedures, J. Am. Stat. Assoc. 69, pp. 86-101.

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

[39] J. Lee, S. Singh, Y. Roh, Interlayer Interactions and Performance in Wireless Ad Hoc Network (draft-irtf-ans-interlayer-performance-00.txt), Internet Draft of IRTF ANS Working Group, September 2003.

[40] 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.

[41] 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.

[42] Michalski, R., Stepp, R. E., and Diday, E., 1983, Automated construction of classifications, conceptual clustering versus heuristic methods for clustering, In Pattern Recognition in Practice, E.S. Gelsema und L.N. Kanal, Eds., pp. 425-436.

[43] P. Michiardi, R. Molva. Game Theoretic Analysis of Security in Mobile Ad Hoc Networks. Research Report RR-02-070, 2002.

[44] P. Michiardi and R. Molva. CORE: A Collaborative Reputation Mechanism To Enforce Node Cooperation In Mobile Ad Hoc Networks. In Proceedings of The 6th IFIP Communications and Multimedia Security Conference, Portorosz, Slovenia, September 2002.

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

[46] Murtagh, F.1984, A survey of recent advances in hierachical clustering algorithms which use cluster centers, Comput. J. 26, pp. 354-359.s

[47] 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.

[48] 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.

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

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

[51] Sudipta Rakshit and Ratan K. Guha. Fair Bandwidth Sharing in Distributed Systems: A Game-Theoretic Approach. Manuscript, 2004.

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

[53] E. Royer, S. Lee and C. Perkins. The Effects of MAC Protocols on Ad hoc Network Communications. *Proc. IEEE Wireless Communications and Networking Conference,* Chicago, IL, September 2000.

[54] 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.

[55] 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.

[56] 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.

[57] N. Schacham and J. Westcott. Future directions in packet radio architectures and protocols. *Proceedings of the IEEE*, 75(1), pp. 83–99, January 1987.

[58] Sneath, P.H. A. and Sokal, R. R. Numerical Taxonomy, Freeman, Lonon, UK, 1973.

[59] V. Srinivasan, P. Nuggehalli, C.F. Chiasserini, R.R. Rao. Cooperation in Wireless Ad Hoc Networks. In Proceedings of IEEE INFOCOM, 2003.

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

[61] 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.

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

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

[64] Ward, J. H. Jr. Hierachical grouping to optimize an objective funtion J. Am Stat. Assoc. 58, pp. 236-244, 1963.

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

[66] Stefano Zanero, Sergio M. Savaresi. Unsupervised learning techniques for an intrusion detection system. Proc. ACM symposium on Applied computing, 2004.

[67] 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).

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