Information Security Journal: A Global Perspective, 17:267–277, 2008
Copyright © Taylor & Francis Group, LLC
ISSN: 1939-3555 print / 1939-3547 online
DOI: 10.1080/19393550802492495

A Structuration Agency Approach to Security Policy Enforcement in Mobile Ad Hoc Networks

Michael Workman,
Richard Ford, and
William Allen
Florida Institute of Technology,
Melbourne, FL, USA

ABSTRACT A mobile ad hoc network (MANET) is a self-organizing, self-configuring confederation of wireless systems. MANET devices join and leave the network asynchronously at will, and there are no predefined client or server roles – roles change based on the nature of a given communication. The dynamic topologies, mobile communications structure, decentralized control, and anonymity creates many challenges to the security of systems and network infrastructure in a MANET environment. Consequently, this extreme form of dynamic and distributed model requires a reevaluation of conventional approaches to security enforcements. Recent developments in agent frameworks have contributed to some potential solutions for security policy enforcements for MANETs. Building on these developments, and extending principles from structuration theory (Giddens, 1984), we formulated a socio-biologically inspired approach to MANET security we refer to as structuration agency theory.

KEYWORDS agency theory, agent frameworks, MANET security, structuration, structuration theory

As part of a U.S. Army Research Office grant-funded project, we developed a biologically inspired security infrastructure (BITSI) for tactical communications. In that phase of our research, we outlined and prototyped an artificial immune system that would allow ad hoc nodes to participate in dangerous environments and receive some damage but continue to operate while self-healing. The analog is of a person who becomes ill and his or her immunology produces antibodies to attack the invading “nonself” pathogens (Hoffman et al., 2008). When ill, a person can develop symptoms, such as a fever, and his or her performance may be degraded, but unless the condition is lethal the person is able to maintain some level of function. In this manuscript, we extend our funded work on individual node or agent immunology and describe how agents are precautionary and detect danger in mobile ad hoc networks, or MANET.

The concept of allowing agents to operate in dangerous settings with the recognition that some agents may suffer some damage goes against the

We wish to thank and acknowledge that this project was funded by a grant from the US Army Research Office, BAAW911NF-07-R-0001.

Address correspondence to Dr. Michael Workman, College of Business, Florida Institute of Technology, 150 W. University Boulevard, Melbourne, FL 32901. E-mail: mworkmanfsu@yahoo.com

conventional wisdom that tries to establish bastions. However, building fortifications depend on fixed sites and predictable configurations (a fortress is not very mobile). Nevertheless, MANET needs to be able to determine if danger is present, and recognize when damage has occurred in order to initiate its artificial immune system. The difficulty presented in MANET is how to facilitate this, given that MANET is a social collection in which one agent at any time may infect another. A key ingredient, in addition of course to having underlying security infrastructure such as firewalls and virus scanners, is to establish an organic security policy enforcement infrastructure.

Codified into information and communications systems, security policies define the rules and permissions for system operations. For example, a router's security policy may permit only egress ICMP messages and deny those that are ingress, or a host computer's security policy may prohibit files from copying themselves, or to access an email address book, or make modifications to the Microsoft Registry.

In conventional systems architecture, security policies are predefined for a given security stance and for a given set of platforms. In these cases, using security policy ontologies have become a popular approach (c.f. Bradshaw et al., 2004) to decouple from some underlying technological interdependencies. In this approach, security policies are predefined using a graphical user interface, and the underlying technology generates the ontology markup (e.g. OWL or DAML), performs policy deconfliction, drops the policies and registers the servers or agents automatically. A web service, agent framework, or object request broker, is then used for policy discovery and enforcement at runtime.

In other cases, security policies may be learned and generated by running an application in a controlled environment to discover its normal behavior, and then subsequently monitoring the application to determine whether it deviates from this predefined behavior. If so, the application execution is intercepted, for example, if it attempts to make privileged systems calls that are prohibited (c.f. Provos, 2002).

While both of these approaches can be extremely effective in many if not most topologies and configurations, mobile ad hoc networks (MANETs) create unique challenges that these more conventional approaches do not address well in isolation, at least in a static configuration (Sterne, et al., 2005). For example, in a MANET, agents exchange information among wireless devices in a flexible topology where devices may join and leave the network unpredictably. Paths between any set of communicating devices or nodes may traverse multiple wireless links, which may be comprised of heterogeneous platforms running various applications and consisting of strict limitations, such as in RAM or in network transmission rates. They also vary in their ability to supply underlying security countermeasures such as firewalls, virus scanners, and intrusion detection systems. Thus these underlying countermeasures cannot be completely relied upon, and to help resolve these MANET problems, building on the extant approaches to security policy enforcement in agent-based frameworks and in peer-to-peer (P2P) networks, we propose a more adaptive socio-biologically inspired approach derived from structuration theory.

The remainder of this manuscript is organized as follows: We first briefly introduce agency and structuration theory and present how these social exchange theories might be applied in a technological environment for the enforcement of MANET security. We follow with an explanation of the agency and social cooperation in MANET security using adaptive stranger policy (Feldman et al., 2004) and biologically inspired artificial immunology (Aickelin and Cayzer, 2002). In so doing, we briefly present ontologies and agentic frameworks for MANET adaptation and security enforcement. We conclude with a description of a use case scenario for MANET security policy enforcement and raise some issues for further study.

AGENCY AND STRUCTURATION THEORY

Agency represents individual behaviors that operate within a broad network of biologic-socio-structural influences (Bjorklund, 1995; Chomsky, 1996). Bandura (2001) defined this phenomenon as "agentic transactions, where people are producers as well as products of social systems" (p. 1). As defined within social-cognitive theory, agency exists on three levels: direct personal agency; proxy agency, which relies on others to act on one's behalf to secure desired outcomes; and collective agency, which is exercised through socially coordinative and interdependent effort (Chomsky, 1996; Bandura, 2001).

Modeling these socio-biological artifacts in software has led to the development of epigenetic systems (Bjorklund, 1995) in which linear models have become supplanted by more dynamically organized computational models that perform multiple operations simultaneously and interactively with the environment in which it operates (Bandura, 2001). Epigenesis from a socio-biological perspective asserts that new structures and functions emerge during the course of developmental interaction between the all levels of the agentic biological and environmental conditions (Bjorklund, 1995). The notion of agency from this perspective contrasts with nondeterministic (chaotic) and nonrational "natural" processes that create the environments in which people operate (Beck, Giddens and Lash, 1994).

Giddens (1984) articulated structuration theory to help explain the role of human agency in the reciprocal relationship between human social systems and their social structures, which Giddens called the "duality of structure." In these terms, structure is defined by the regularity of actions or patterns of behavior in collective social action, and agency is the human ability to make rational choices and to affect others with consequential actions based on those choices.

Structuration is therefore a mutually binding dynamic activity that emerges from the social interaction. More specifically, social action relies on social structures, and social structures are created by means of social action. Thus structures derive the rules and resources that enable form and substance in social life, but the structures themselves are neither form nor substance. Instead, they exist only in and through the activities of human agents. For example, people use language for communications with one another, and language is defined by the rules and protocols that objectify the concepts that people convey to each other (Chomsky, 1996).

The syntax structure of language is the arrangement of words in a sentence. By their relationships of one to another (e.g., subject-predicate noun-verb phrase), the sentence structure establishes a well-defined grammar that people use to communicate. However, language is also generative and productive and an inherently novel activity, allowing people create sentences using the syntax rather than to simply memorize and repeat them (Chomsky, 1979). Furthermore, social structures such as rituals, rites of passage, and ethical codes of conduct, are dynamic and can change when they no longer serve the purpose for which they were created. People negotiate the effects of these social structures and collectively begin to ignore them, replace them, or reproduce them in new forms if they become dysfunctional (Beck et al., 1994).

Equivalent to the conventional static approaches to security policies in a technological environment, policy definitions are akin to memorizing sentences and are rigid and non-negotiable. However, a mobile ad hoc network (MANET) is self-organizing, self-configuring, and systemic, where wireless systems may join and leave the network asynchronously and unpredictably (Sterne et al., 2005). Since there are no predefined client or server roles, the ability to negotiate collective social action is important to the simultaneous demands for structure that enables agents to establish and cooperatively carry out goals as well as the flexibility to create those structures, in other words, to support the *duality of structure*.

Analogous to human agentic transactions, in MANETs, agency acts on three levels:

1. Direct personal agency, in which an agent has goals, makes plans, and takes steps that are governed by certain rules (analogous to agentic biologically driven behaviors),
2. Proxy agency, which relies on others to act on one's behalf to secure desired goals (analogous to agentic sociologically driven behaviors), and
3. Collective agency, which is conducted through socially cooperative and interdependent efforts that affect other agents within a social network (epigenetic behaviors).

By way of illustration, in a biological analog, designers of inorganic systems have relied on lessons from human immunology to develop genetic algorithms and self-healing systems (Aickelin and Cayzer, 2002). In a sociological analog, digital sociologists have relied on principles from social networking that have led to developments such as viral marketing and incentive-based cooperative systems (Feldman et al., 2004).

Taken together, these have inspired modeling agentic security transactions in MANETs, where agents behave socially to exchange information, receive instructions, react to the effects of other agent actions, and provide responses in a cooperative fashion to fulfill individual and collective goals in an adaptable and

TABLE 1 Agentic attributes

Autonomy:	The ability to pursue its own individual set of goals and make decisions by monitoring events and changes within its environment.
Proactivity:	The ability to take action and make requests of other agents based on its own set of goals.
Reactivity:	The ability to take requests from other agents and react to and evaluate external events and adapt its behavior and make appropriate decisions to carry out the tasks toward goal achievement.
Social cooperation:	The ability to behave socially, to interact and communicate with other agents in multiple agent systems (MAS) to achieve collective goals.
Negotiation:	The ability to conduct organized conversations to achieve a degree of cooperation with other agents.
Adaptation:	The ability to improve its performance over time when interacting with the environment in which it is embedded.

evolutionary way, while simultaneously healing from and warning others of security violations and violators (Boella, Sauro, and van der Torre, 2005). We refer to this as a *structuration-agency* approach to MANET security (see Table 1 for agentic attributes). In the next section we posit an implementation of structuration agency with agentic security transactions on the three levels presented.

AGENCY AND SOCIAL COOPERATION IN MANET SECURITY

Direct Agency

For agents to take individual and social action, they must rely on the information they have to make predictions about the consequences of their behavior. In other words, to be proactive, agents require an initial set of knowledge from which agents base their assumptions (Dastani et al., 2005). This knowledge may be stored as a set of ontologies, which are methods for organizing and sharing information derived from a common understanding of the structure of information (Gruber, 1993; Musen, 1992). Security policies constitute domain ontologies, whereas objects, events, rules, and processes involved in the enforcement of security policies are foundational ontologies. Thus the foundational ontologies provide the structures that comprise the rules and resources (sets of transformational relations) and the properties that define the agent, and that establish the patterns of agency behavior.

In relation to security, at the direct personal agency level, a base set of information is also needed for an agent to, in Aickelin and Cayzer's (2002) terms, distinguish self from nonself. It is important to note that not all nodes participating in a MANET will possess this ability, but the efficacy of the agent's immune response system will be enhanced by these data because they are crucial to the anomaly detection process in the immunology. The discussion of artificial immunology is beyond the scope of this manuscript; however, without this capability, the agent must rely solely on underlying security infrastructure such as confederated intrusion detection systems (Sterne et al., 2005) for its defenses.

While confederated intrusion detection offers some clear benefits in MANET security, the administrative capability to establish such a topology is not always possible. Something more is needed, and to the extent that an agent can understand and manage itself in the MANET, the more autonomously it can act (i.e., it relies less on a hierarchy of structures or administrative domains), and when critical information about self is unknown, it is learned, and this learning is most likely to be derived socially. In Figure 1, we present a high-level ontology used for self-definition.

As seen in the ontology representation, an agent defines self in terms of its configuration. A configuration rule governs actions to the self and is composed of an agent's identity. Identities have one or more roles, and depending on the permissions allocated to the role (in an RBAC sense) actions may or may not be taken with the configuration. Actions relate to the configuration such as administrative rights to install software or start a service. It is important to note that in MANET, not all nodes will have this capability – indeed, some agents may be little more than receiver-transmitters, but the higher value nodes will likely have incrementally more capabilities to support these features.

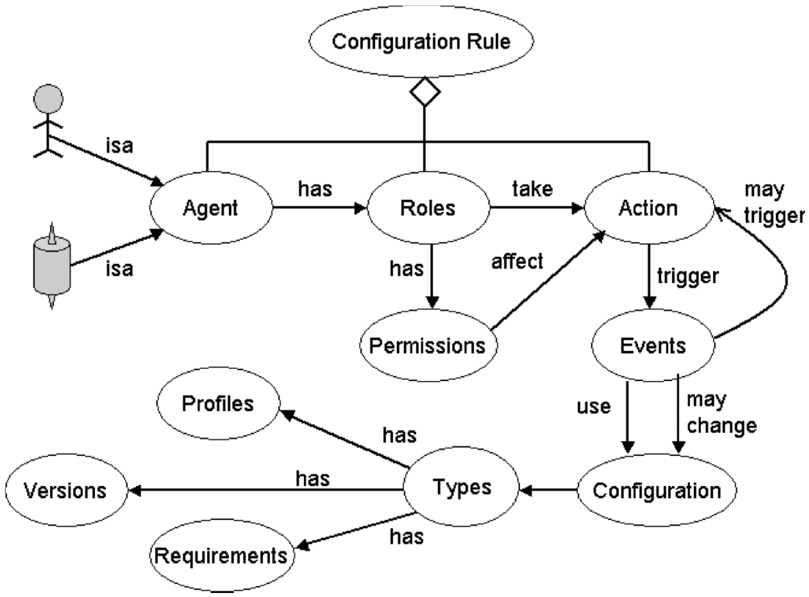

FIGURE 1 Configuration Ontology Representation.

Depending on the flexibility needed for the agent's role, the ability to use and or make changes to the configuration may be left "wide open," similar to anonymous logins. There are cases in certain field settings where some devices will need to allow anyone who has access to a MANET agent to install software, start services, or otherwise change the configuration. Higher value nodes will not likely permit this, and the graduating agent infrastructure and ontology will allow these types of decisions to be made according to the needs.

An action may lead to other actions depending on the events the actions trigger. Moreover, depending on the authorizations, actions may permit usage of elements in the configuration, or even to change the configuration if authorizations allow it. Configuration types have profiles, which can be as simple as file sizes, access permissions, and check sums. In the most robust and stringent sense, a profile may consist of all the normal behaviors a given application can perform including the system calls that the application makes (c.f. Provos, 2002). A type of configuration may also have a version associated with it, as well as requirements, such as the amount RAM necessary for execution or a set of Quality of Service (QoS) metrics to fulfill its obligations.

Proxy Agency

MANET autonomy is at odds with cooperation. For example, a phenomenon coined by Feldman and Chuang (2005) as "selfish link" reflects agent actions that lead to a lack of cooperation and undesirable network effects. Among these issues is "free riding" (Feldman and Chuang, 2005) where agents disregard their obligations to other agents in favor of self-preservation, for example, to preserve its own compute cycles or communications bandwidth for its own services, such as running local services at the highest priorities and lowering priorities of requests from external nodes. To address these problems, incentive models may be used to encourage more altruistic behaviors.

An incentive mechanism often used successfully in P2P networks has been reciprocity (Antoniadis et al., 2004). Reciprocity is the condition in which one node will agree to service a requestor (even if the there is no preexisting obligation to the requestor) in exchange for consideration *quid pro quo*. However, reciprocity alone is inadequate to address MANET proxy agency behavior, as we shall explain shortly, and moreover, the technique requires agents to maintain histories of interactions with requestors. Among the issues that relying on these incentives present include the fact that private histories fail when the nodes in the network reach a critical mass, and they allow if not encourage collusion and "whitewashing" security threats (Lai, Feldman, Stoica, and Chuang, 2003). Collusion occurs when multiple nodes report positive experiences with a malevolent agent to other nodes. Whitewashing occurs when a node leaves the MANET and rejoins with a different identity to avoid its negative history with other nodes.

For an individual agent to elicit support in completing its desired goals it must trust its proxies and vice versa. Yet the ad hoc nature of the MANET topology lends toward distrust, and distrust in conventional security policy enforcement may lead to strict constraints or even preclusion of participation by new nodes ("strangers") joining the network (Feldman et al., 2004). Because of this, MANET administrators are often tempted to rigidly define security policies and establish a pessimistic security stance. However, this can create suboptimal and even dysfunctional MANET performance because in ad hoc environments such as MANET, cooperative actions can bring indirect and intangible returns to agents (Boella et al., 2005).

A stranger in MANET systems may be crucial to successful MANET operations because peer nodes in the MANET may share an obligation to exchange or forward new communications based on an individual agent's environment or experience (i.e., events and histories that it logs). For instance, if all agents generously shared their resources, each agent would be able to contribute to carrying out the mission of the MANET, such as creating high availability communications throughout the MANET fabric. In other words, social cooperation is needed for individual agent's goal pursuit. Resources must be shared among the agents in a MANET since it is (typically) the collective effort of the agents that will lead to the achievement of the established MANET objectives.

Collective Agency

There are three behavioral approaches to collective agency: providing incentives, such as those given to agents to cooperate; negative reinforcements, such as those to cause an agent to follow through on an obligation or to cease malevolent or unintentional damaging behavior; and punishments levied against agents that ignore negative reinforcements or that are violators of a security policy or issue a request that is a known threat. All three approaches are needed to form robust agentic security policy enforcement, but each behavioral approach targets a different set of conditions, and security policy enforcement must make proper distinctions about when and where to apply which technique. In Figure 2, we present a high-level epigenetic ontology for collective agentic behavior.

In the epigenetic ontology, we first note that it imports the Common Vulnerabilities and Exposures (CVE) ontology (see Moreira et al., 2008, for a description). The description of that ontology is beyond the scope of this manuscript, but it captures and updates with common vulnerabilities and incidents such as reported by the Software Engineering Institute's CERT.

The primary function of the epigenetic ontology is to define a set of standards, criteria, and behaviors that direct the efforts of the agents toward cooperative goal achievement. Consequently, the agent assumes an optimistic security stance and allows requests even if there is no prior obligation, so long as the request is not explicitly prohibited or is associated with a known vulnerability threat. This of course is regulated by the configuration rules, and supports graduated security depending on the role and needs of the agent.

While making such allowances and offering incentives for doing so are helpful in gaining cooperative behavior such as to entice selfish agents to participate in routing and forwarding of data (Kwok, 2007), they do not discourage malevolent behaviors such as nodes that continue transmissions even after ICMP source quench requests are made (Buchegger and Le Boudec, 2005). From this example, danger then might be inferred if an agent issues a notification that is subsequently ignored.

Hence, the inclusion of a reputation-based system mitigates potential hazards by maintaining and collecting votes from other agents about their favorable or unfavorable history with the requestor. If the requestor has an unfavorable reputation as determined by the agent's local history, an agent may resort to punishing the malevolent requestor by simply adding it to its service prohibitions. On the other hand, if the other agents report an unfavorable reputation, but an agent's local history is favorable, it may choose to allow but monitor the behavior of the request such as its consumption of the agents resources (CPU, bandwidth, memory) or attempt to make changes to the configuration such as copying, modifying, or deleting a file.

If an agent with a good reputation begins to cause damage to the system, such as through requests of a provider that absorb all available resources, the agent may issue negative reinforcement demands that the

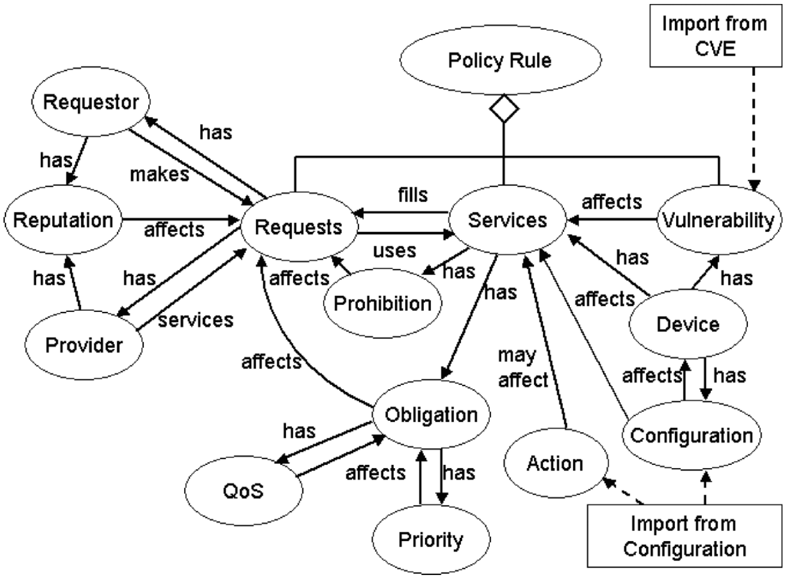

FIGURE 2 Epigenetic Ontology for Collective Agency.

requester reroute or throttle its requests, and then monitor for compliance and update the reputation history and report to other agents accordingly. If the requestor ignores the demands, the agent may switch to punishments.

A request may be made from an agent with no prior history. It such cases, the "stranger" may have left the network and rejoined with a different identity (a whitewash). In addition, there exists the possibility of collusion among agents that may give a malevolent agent a high reputation value. This can be addressed in a couple of ways. First, Sterne et al. (2005) suggested the use of a dynamic hierarchical model in which nodes and services are partitioned up through to an authoritative root. The system relies on clustering to enable scalability, resilience (fault-tolerance), and adaptability to the dynamic environments in MANETs. Each node in the peer group maintains responsibility for its own security (from an IDS perspective), as well as some limited responsibility for its adjacent peers, which is summarized and propagated up the authoritative chain. Directives, on the other hand, are passed top-downward.

Another technique is through an adaptive stranger policy (Feldman et al., 2004; Lai et al., 2003), which deals specifically with novel behaviors and therefore is useful in terms of assessing danger prior to damage occurring to an agent. Instead of penalizing a new user (which would discourage cooperation among peers), the adaptive stranger policy requires that each existing peer, before deciding whether to share a transaction with a new user, computes a ratio of the amount of services provided to amount of services consumed by a new user. If the ratio is great than or equal to 1, then the existing peer will work with the new user. On the other hand, if the ratio is smaller than 1, then the ratio is treated as a probability of working with the new user (Sun and Garcia-Molian, 2004).

MANET SECURITY ENFORCEMENT AS DANGER AND DAMAGE CONTROL

From a security policy perspective, an agent is granted rights according to a given role, but in an *ad hoc* environment rights and roles can be dynamic. Behavioral role conformance to a well-defined set of behaviors based on rights represents the usual case, and is benign. Benign behaviors do not need to consume system resources to closely monitor. However, when something novel is encountered, it presents potential danger. Novelty may be such that an agent attempts to perform an unauthorized function, or an agent performs an authorized function, but the function attempts to perform some novel behavior to be monitored. Danger, therefore, can equate to novelty in terms of security policy enforcement.

From this frame of reference, danger can also be viewed as a continuum on a severe (X coordinate) and imminent (Y coordinate) grid. When danger is encountered, it is monitored according to its coordinates on this grid. A threshold can be set in the continuum in which an intervention may be required to preclude damage from occurring. Damage in this context may be defined as any action that would impinge on mission execution, including negative effects on mission parameters such as exponential consumption of network bandwidth, an application that normally does not copy files tries to copy a file, or negative impact on any QoS parameters needed for successful mission execution.

The structures of the agentic actions consist of goals, plans, steps, and rules, which are malleable and negotiable. That is, an agent assembles its own set of rules dynamically while reacting to events and selecting appropriate qualifying plans to achieve its current goal. The agent may try other qualifying plans if the initial plan fails, or if events or exceptions cause the agent to change plans. Consequently, agents are adaptive to a given situation in that they select the next plan based on current information that is available, either from the environment or from its local knowledge (Dastani et al., 2005).

The goal is the root of the agentic hierarchy, and it consists of the agent's obligations and prohibitions. As a case in point, a goal might be to obligate the agent to provide a Web service. Goals consist of plans to execute in order to satisfy the goal; for example, an obligation to provide a Web service might require a plan to start httpd on port 8080. Agentic behaviors are governed by rules, which might specify that an ActiveX control is not permitted into the Web service. Rules operate on the steps that are part of a plan; thus, if an agent receives an http request containing an ActiveX control, the rule may require steps to discard the request.

The choreography of agentic actions may be decomposed into three levels: high, intermediate, and low. Perhaps an agent has multiple goals, and each goal has multiple plans. For example, the goal "maintains current version levels of applications" and may have a high-level plan entitled "Version Updates." For this high-level plan, there are intermediate plans that perform a sequence of steps, tasks, and log data updates, such as "Automatic Update AcroRd32.exe" may require a network connection to be opened and a file download from the Adobe Website over a TCP/IP socket. Low-level plans perform the system tasks and log data updates, for example, open 127.0.0.1:2076, and record the conversation with Adobe (Sse Table 2 for an example hierarchy of goal pursuit).

TABLE 2 Agency goal pursuit

Agent → Goal → Goal → 0
Goal → Plan → Step → Plan → Step → Plan → 0
Step → Rule → Step → Rule → Step → 0

This way, beginning with an initial set of plans, execution may proceed through paths from one plan to another by the agent, and this allows it to limit the scope of its response to address more localized problems (e.g., provide an http connection for agent X, but not for agent Y). Also, if a goal can be achieved in different ways (e.g., manual or automatic update), the three levels of plans allow for this localized ability.

Damages that may occur vary according to the types of activities that an agent attempts. In data sharing for example, agents need to utilize at least two different forms of resources: storage in which each agent has to set aside some storage space for files that may be needed by other agents even though these files may not be useful to the agent itself, and bandwidth where each agent must devote some of its bandwidth for messaging and communications requested files by other agents. Damage in this specific sense is assessed according to the excess of security policy-defined thresholds. Figure 3 shows MANET security policy enforcement for the use case scenario described in Table 3.

To illustrate the use case scenario, MANET agents interrogate their configurations and vulnerabilities ontologies against requests for services according to available resources, access and rights, reputations, and "suitable behaviors." Optimally, as recommended by Provos (2002), running initially through applications and generating profiles could create baseline configurations, but this is not always possible. It is more likely in MANETs to compromise by learning and adding benign behaviors into the agent's ontology. For example, a request may be made for Web service on port 8080. When the request is serviced, the agent checks the ontology for the Web service to determine what actions are permissible. As long as the requestor behaves properly, no danger is detected, and permissions are granted for using resources according to defined QoS (e.g., memory utilization, network utilization, CPU utilization) and configuration (e.g. file checksums) and plan parameters.

In order for the agent to initiate a self-healing process after servicing a request that causes damage, for

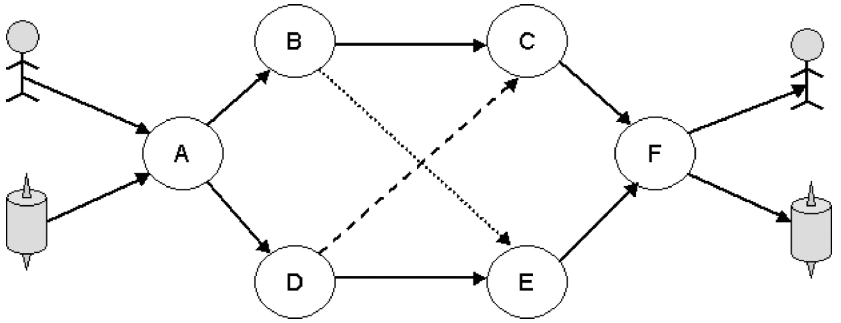

FIGURE 3 Enforcement Use-Case Scenario.

TABLE 3 Use case scenario: message relay

Relay message tactical level:
1. Sender creates message
2. Message is addressed to receiver
3. Message routed from sender to receiver through MANET

Relay message operational level:
1. Relay makes request of provider for service; pass request, reputation
2. Provider checks service for prohibition
3. Provider checks device for vulnerabilities
4. Provider checks service for obligation; receives priority and QoS

Alternative 1: Request prohibited
Alternative 2: Service priority preempted
Alternative 3: Service unable to meet QoS
Alternative 4: Service vulnerable to threat from request

1. B has high priority obligation to A, A requests B, exceeds QoS threshold Agentic Action: Reply A to throttle or reroute requests
2. D has medium priority obligation to A, A requests D Agentic Action: Accept request, and queue request according to priority
3. C has no obligation to D, D has good reputation, D requests C Agentic Action: Accept request and queue below obligation priorities
4. E prohibits B, B request E Agentic Action: Deny Request (drop request)
5. F has high priority obligation to C, C has good reputation, C requests F Agentic Action: Accept request and queue request according to priority
6. F has no obligation to E, E has poor reputation, E requests F Agentic Action: Accept request, queue below obligation priorities, quarantine and monitor Action and Configuration.

instance, if an agent detects violations to QoS mission parameters or a violation of normal behavior, it calculates a vector for the violation and determines damage severity. If severe damage is determined after a servicing a request, the damage is flagged as malignant self and filtered through a damage controller to determine what part of self is malignant and what actions to take, and the agent whose request caused the damage is added to the prohibitions and given a low reputation.

On the other hand, if a requestor is a stranger, or if a requestor with a high reputation makes a novel request, then danger is determined and the request is proactively monitored. That is, in cases of agent or behavior novelty, the request is quarantined and monitored to determine what resources it may utilize and to what extent, according to the obligation QoS parameters, or available resources in lieu of obligations. If a threshold is likely to be exceeded or a resource violation is likely to occur, the agent may notify the requestor, for example, to cease or reroute the request. If the requestor complies, a high reputation is given. On the other hand, if the request represents a known threat, the request is flagged as malignant nonself, and is filtered through a damage controller to determine what actions to take, such as to deny the request, and issue a low reputation for the requestor.

As suggested in the use case scenario, agents are permitted to perform actions not explicitly denied them, and dangerous behavior is monitored but not prevented. Because of their nature, MANETs cannot predefine all possible behaviors for agents *a priori*. A baseline can be established, but there may be occasions when agents legitimately need to perform novel behaviors. An agent monitors its environment by acquiring information from other agents, which include reputations of other agents, and monitor self according to QoS metrics, obligations, system resources and utilizations, and so forth. Given a goal and information about its situation, and information about its environment, an agent selects a plan and executes the plan's defined steps.

When an event occurs, depending on the rules governing the behavior, the agent may suspend its current plan and choose a new one. For instance, if an agent has a goal of providing a service and receives an update indicating a poor reputation of a peer, the agent may suspend its planned obligation for providing the agent the service and select a new plan. Thus, an agent selects a plan that fits the goal based on its environment and immediate situation. In summary, although social and cooperative, agentic actions work similarly to the body's natural immune system, the idea behind this is to allow danger and the potential for some damage, but to ensure survivability of the agent so that it can continue to provide cooperative support to the collective agency in the MANET.

CONCLUSIONS AND IMPLICATIONS

Statically defined security policies and configurations are insoluble in mobile ad hoc network (MANET) environments. A more practical solution to carrying out operations in MANET is to facilitate direct personal agency, enabling the flexibility for agentic goal-directed autonomous behavior. However, given the nature and limits of the topology, proxy cooperation with other agents is essential (Boella et al., 2005). Each agent carries out its own set of goals according to its plans and makes requests of other agents, which are fulfilled so long as there remains cooperation and "good" behavior. A plan consists of the sequence of steps associated with the goal and rules that govern its responses to requests, events, and exceptions. An agent may have multiple plans available for a given goal, and multiple goals to accomplish, which are dynamic and negotiable (Dastani et al., 2005).

Collective agency forms in response to agent reactions to events and other agent requests, and agents provide services based on its available resources and the agreements the agent forms socially in the MANET. However, the combined sociability and the ad hoc nature of the MANET create an insecure environment. It must be possible for an agent to have the ability to detect severe impending danger so that some preventative measures can be taken before damage occurs. Malevolent behavior is contained and reported to other agents by reputation, and some conservatism is built into the MANET by treating strangers cautiously but not denying them the opportunity to prove themselves as trustworthy.

In this manuscript, we outlined a set of ontologies and illustrated a use case for agency and MANET security based on structuration theory. The self-healing aspect of agency in a structuration framework has yet to be fully explored. Dangerous behavior can occur from both legitimate agents (those who are authenticated) and those who are intruders. Dangerous behavior by legitimate agents may involve executing novel behaviors that might affect mission parameters or QoS metrics.

On some level, danger needs to be monitored but not necessarily disrupted, unless it presents a severe and imminent threat potential for lethal damage. However, dangerous behaviors by undetected illegitimate agents (or colluders) may involve eavesdropping or interception of communications and may not escalate to disrupting the mission parameters and might not be detected, and thus rely on the typical cryptographic approaches to this problem –which may not be supported by all MANET agents.

Another challenge in MANET applications that needs to be addressed is the establishment and monitoring of QoS parameters. Mission specific parameters and QoS metrics become crucial in determining the distinction between danger (novel behaviors) and actual impacts (damage) on mission execution. Applications may consist of discrete actions such as file or message exchanges, or involve streaming data such as video or voice. In discrete actions, the entire message or file may need to be received before it can be examined by an agent causing delay between the time of the service request and the assessment of its quality. In an extreme case, an agent may not discover that a transferred file is valid or is a dangerous file until the file has been completely loaded into the agent. Streamed data on the other hand allows an agent discover if the data are acceptable during points in the transmission.

In addition, the QoS metrics may be different between discrete and stream data. In cases such as file transfers, metrics are generally download time and integrity of the files transferred. With streaming data, more crucial performance parameters tend to be the degree of jitter, frame rate transfer, and resolution. Danger provokes monitoring – and because in an ad hoc environment, novel behaviors must be tolerated until it is learned whether the actions are benign or malignant but does not necessarily preclude them – depending on the assessment of the severity and imminent threat assessment. Damage on the other hand, cannot be tolerated (at least for long). Being able to rapidly assess the QoS and timely actions by an agent is critical problem to solve.

Finally, another area of interest for further exploration is in malevolent agent discovery. Since security policy ontology rules can be structured as subject → event → object, it should be possible to detect collusion issues among both cooperative and noncooperative agents that may take place, for example, by correlating subjects with events, and events with objects, and subjects with objects to look for collusion patterns. However, by the time this can be done, the colluders may have disappeared from the MANET or have whitewashed. Research into ongoing techniques and remains a yet-to-be conquered challenge.

REFERENCES

Aickelin, U. and Cayzer, S. (2002). "A open-source" working paper: The danger theory and its application to artificial immune systems. The proceedings of University of Kent at Canterbury, 1, 141–148. Retrived 01/11/08 from: http://www-uk.hpl.hp.com/people/steve_cayzer/Publications/ICARIS_danger_final.pdf. The danger theory and its application to artificial immune systems, *University of Kent at Canterbury*, pp. 141–148.

Antoniadis, P., Courcoubetis, C. and Mason, R. (2004). Comparing economic incentives in peer-to-peer networks, *The International Journal of Computer and Telecommunications Networking*, 46, pp. 133–146.

Bandura, A. (2001). Social cognitive theory: An agentic perspective, *Annual Review of Psychology*, 52, 1–26.

Beck, U., Giddens, A. and Lash, S. (1994). *Reflexive modernization. Politics, tradition and aesthetics in the modern social order*. Cambridge: Polity Press.

Bjorklund, D. F. (1995). *Information processing approaches: An introduction to cognitive development*. Washington, DC: Brooks-Cole.

Boella, G., Sauro, L. and van der Torre, L. (2005). Admissible agreements among goal-directed agents. *Proceedings of the IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT'05)*. March 2005, Paris, France.

Buchegger, S. and Le Boudec, J. Y. (2005, July). Self-policing mobile ad hoc networks by reputation systems, *IEEE Communications Magazine*, pp. 101–107.

Chomsky, N. (1979). Human language and other semiotic systems, *Semiotica*, 25, pp. 31–44.

Chomsky, N. (1996). *Language and problems of knowledge*. Mendocino, CA: MIT Press.

Dastani, M., van Riemskijk, M. B., Dignum, F. and Meyer, J. J. (2004). *A programming language for cognitive agents goal directed 3APL*. *Lecture notes in computer science*. Heidelberg: Springer, pp. 1611–3349.

Feldman, M. and Chuang, J. (2005). Overcoming free-riding behavior in peer-to-peer systems, *ACM SIGccom Exchanges*, 5, pp. 41–50.

Felman, M., Lai, K., Stoica, I. and Chuang, J. (2004). Robust incentive techniques for peer-to-peer networks, *Proceedings of the 5th ACM Conference on Electronic Commerce*, *102–111*. New York: Communications of the ACM.

Giddens, A. (1984). *The constitution of society: Outline of the theory of structuration*. Cambridge, UK: Polity Press.

Hoffman, K., Ondi, A., Ford, R., Carvalho, M., Brown, D., Allen, W. H., & Marin, G. A. (2008). Danger theory and collaborative filtering in MANETs. *Journal of Computer Virology* 10: 1772–1794.

Kang, S.S. and Mutka, M.W. (2005). A mobile peer-to-peer approach for multimedia content sharing using 3G/WAN dual mode channels, *Wireless Communications and Mobile Computing*, 5, pp. 633–647.

Krishnana, R., Smith, M.D., and Telang, R. (2003). The economics of peer to peer networks, *Journal of Information Technology Theory and Application*, 5, pp. 31–44.

Moreira, E., Martimiano, L., Brandao, A. and Bernardes, M. (2008). Ontologies for information security management and governance, *Information Management & Computer Security*, 16, pp. 150–165.

Sterne, D., Balasubramanyam, P., Carman, D., Wilson, B., Talpade, R., Ko, C., Balupari, R., Tseng, C-Y, Bowen, T., Levitt, K. and Rowe, J. (2005). A general cooperative intrusion detection architecture for MANETs, *Proceedings of the Third IEEE International Workshop on Information Assurance (IWIA)*. April 2005, Washington D.C., pp. 57 – 70.

Sun, O. and Garcia-Molian, H. (2004). SLIC: A selfish link-based incentive mechanism for unstructured peer to peer networks. Proceedings of the 34th International Conference on Distributed Computing Systems. June 2004, Los Alamos, CA: IEEE Computer Society.