Network Meta-Reasoning for Information Assurance in Mobile Agent Systems

Donovan Artz Max Peysakhov William Regli
Department of Computer Science
Drexel University
Philadelphia, PA 19104-2875

1 Introduction

This paper develops a practical means of measuring information assurance for mobile agent systems operating on wireless, ad hoc networks based on meta-reasoning [Dix *et ai,* 2000; Xuan *et al,* 2001] to improve the security of communication. Figure 1 shows an agent system and its two distinct layers of communication: host-to-host and agent-to-agent. Given the plethora of new techniques for identifying network intruders, we study the *compromised host* problem: determining the appropriate response to an identified intruder. In the context of a mobile, multi-agent system operating on an ad hoc network [Forman & Zahorjan, 1994], it is not merely a simple matter of removing the compromised hosts and its agents. While keeping the compromised host can result in *information disclosure,* removal of the host can degrade or even sever the network. Wc develop a state description for an agent system and introduce a measure of* *information assurance* for the system in terms of the integrity of the messages delivered to the agents in a given network state. Agents have three responses to a compromised host: *ignore* the compromised host; *reroute* around the compromised host using network *route redundancies;* or *remove* the compromised host, by having the agents instruct their hosts to eliminate it from the network. These responses are shown in Figure 2.

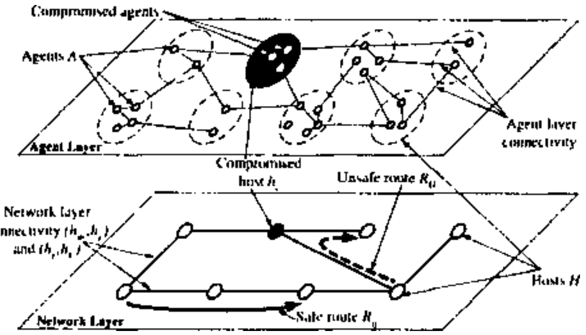

Figure 1: Nodes are either physical hosts or agents; edges are network host or agent-to-agent connections. The shaded node is a *compromised host.* The bold arrow is a *safe network route;* the dashed arrow is a *compromised network route.*

2 Technical Formulation

A state description for a *mobile agent network* is defined in terms of sets of hosts (H) and agents (.4). Given *H* and A, we can define:

- ***Agent topology,*** T_A, is a graph of the connections between agents in the agent layer: $T_A = (A, L_A)$ where $L_A \subseteq A \times A$.
- ***Host topology,*** T_H, is a graph of the direct connections between hosts in the network layer: $T_H = (H, L_H)$ where $L_H \subseteq H \times H$.
- ***Host routing,*** $\mathcal{R}$, is a set of sequences of hosts which describes the network routes used between hosts in the network layer: $\mathcal{R} = \{R_{i,j} : h_i, h_j \in H\}$ where $R_{i,j}$ is the current physical network route from host h_i to host h_j. If no route exists from h_i to h_j, then $R_{i,j} = \emptyset$.
- ***Agent location,*** η, is a function mapping an agent to the host on which it is physically located in a given state: $\eta : A \rightarrow H$. Inversely, the function η^{-1} maps a host to the set of agents located on that host.

The state of a mobile agent network is defined as a tuple: $N = \langle T_A, T_H, \mathcal{R}, \eta \rangle$

2.1 Evaluation of an Agent System Network

We model the information assurance level in an agent network by analyzing how agents communicate. Observe that agents must send messages to other agents in order to collaborate in any decision procedure. In a decision procedure, typically certain agents are *authorities* that collate voting messages from all other agents involved in the decision. In the context of T_H, any host housing an agent involved in at least one decision procedure authority is a *sink* in the network topology graph into which messages flow. For a simple decision, there may be only one sink; in the most complex case, all hosts are sinks. This paper will assume the most complex case, in which all hosts are housing decision authority agents.

For any decision procedure, messages sent to the decision authority can be classified into *successful messages* and *failed messages. A successful message* is delivered without using the compromised host. *A failed message* either: (1) originates or ends at the compromised host; (2) is routed through the compromised host; or (3) cannot be routed because no network route exists.

For a given TV, a change in 7// can could cause a change

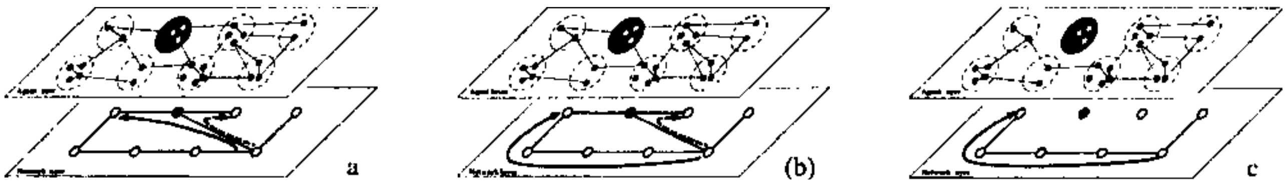

Figure 2: Responses to compromised hosts: (a) *ignore,* (b) *reroute;* (c) *remove.*

in routing. If a route changes, the time taken to transmit a message over that route may also change. A change in message delivery time can negatively impact a decision procedure. Moreover, a compromised host contains agents that may violate their expected behavior in a decision procedure. Both factors must be considered when evaluating a state TV. Wc define two values that can be used to evaluate TV with respect to each factor:

1. A *message integrity rating,* which relates successful messages received to failed messages.
2. A *time rating,* which is an estimate of optimality for the current network routes.

Measuring Message Assurance. If each agent is in communication with all of the other agents, $n_i = (|A| - |\eta^{-1}(h_i)|)|\eta^{-1}(h_i)|$ is the total number of messages that the agents on host *hi* expect to receive from agents on other hosts per unit of time given state TV. Based on the current routes $\mathcal{R}$, one can calculate the number of successful messages received on host h_i, $succ(h_i)$. The *message integrity rating* for host h_2 is computed as: $m_i = \frac{succ(h_i)}{n_i}$. Note, as *mi* increases, the integrity of messages sent to host *hi* also increases. When all messages sent to h_i are successful, mi= 1.0. When all messages sent to h_i fail, m_i = 0.0. The mean *message integrity rating* over the entire mobile agent network in state TV is: $\bar{m} = \frac{\sum_{i=1}^{|H|} m_i}{|H|}$

Measuring Network Routing Efficiency. The trade off is between message integrity and the timeliness of message delivery. Network routing algorithms find sets of routes that minimize some value for all routes in a network. In general, routing algorithms use a weight function on each route and find the shortest, single source, paths to all vertices in T_H. In this context, the function *p* represents the network routing algorithm which returns a set of shortest path routes for the set of hosts, given their current physical network topology: $\mathcal{R} = \rho(H, L_H, w)$, where *w* is the edge weight fiinction. There are several schemes that can be used to weight routes in wireless, ad hoc networks [Royer & Toh, 1999], all of which can be approximated or bounded using a w that returns a value proportional to the time required to transmit a message through the network. The units of time returned by *w* are used for relative comparison of network routes, which we normalize *w* to simplify computations: *w :TZ* → [1, *\H*|], where $w(R_{i,j})$ = |H| signifies that the route from host h_t to h_3 is non-existent and weight of the longest possible route is *\H\ — 1.* Now we can define a *time rating, Ti* for the network routes used by host $h_\%$ as: *ti* = $\frac{\sum_{j=1, j\neq i}^{|H|} w(R_{j,i})}{|H|(|H|-1)}$ Note, as t_t increases, the routing efficiency to host *hi* decreases. When the routing efficiency is minimized (i.e., no connections exist) for a host h_i, t_i = 1.0. As the routing efficiency increases (i.e., shorter routes are used) for host *hi, ti* approaches 0.0. Hence, the mean *time rating* for the entire mobile agent network in state N is: $\bar{t} = \frac{\sum_{i=1}^{|H|} t_i}{|H|}$

Assurance for Whole Mobile Agent Network. A linear combination of *message integrity rating* and *time rating* defines a utility function assessing a mobile agent network in state TV in terms of both assurance and routing efficiency:

$$V(N) = \frac{\alpha m - (1-\alpha)\bar{t} + 1}{2} \qquad (1)$$

a is a coefficient between 0 and 1 that determines the balance between assurance and network performance. If α — 1.0, only message integrity is considered; if α — 0.0, only time efficiency is considered. Note that *V* : TV → [0.0,1.0], where V(TV) = 1.0 is the best possible result.

2.2 Operators on an Agent System Network

Using Equation 1, agents can decide how to operate on their network. Naturally, the *ignore* operator does nothing. The *reroute* operator generates a new set of network routes, using only safe routes wherever possible. If *VR'*is the set of *safest possible routes,* the resulting mobile agent network $N' = \langle T_A, T_H, \mathcal{R}', \eta \rangle$ is generated using the reroute operator on TV (Figure 2(b)). Given the routing algorithm *p* and route weight function *w,* the following algorithm can be used to compute TV';

```
Algorithm 1    reroute(N,h_c)
    $\mathcal{R}' \leftarrow \rho(H - \{h_c\}, L_H, w)$
    $\mathcal{R}' = \mathcal{R}' \cup \{R_{i,j} : R_{i,j} \in \mathcal{R} \wedge R'_{i,j} \in \mathcal{R}' \wedge R'_{i,j} = \emptyset\}$
    $return(\langle T_A, T_H, \mathcal{R}', \eta \rangle)$
```

The *remove* operator results in the complete removal of the compromised host from participation in the agent system's underlying network. The new mobile agent network, $N'' = \langle T'_A, T'_H, \mathcal{R}'', \eta \rangle$, is the result of applying the removal operator on TV and is generated by the following algorithm:

```
Algorithm 2 remove(N, h_c)
    $A' \leftarrow \{a : a \in A \wedge \eta(a) \neq h_c\}$
    $L'_A \leftarrow \{(a,b) : (a,b) \in L_A \wedge a, b \in A'\}$
    $T'_A \leftarrow (A', L'_A)$
    $H' \leftarrow H - \{h_c\}$
    $L'_H \leftarrow \{(h_i, h_j) : (h_i, h_j) \in L_H \wedge h_i, h_j \in H'\}$
```

```
$T''_H \leftarrow (H', L'_H)$
$\mathcal{R}'' \leftarrow \rho(H', L'_H, w)$
$return(\langle T'_A, T'_H, \mathcal{R}'', \eta \rangle)$
```

In order to select the operator resulting in the highest valued agent system, consider the values *V{N), V(N'),* and *V(N").* The highest of these values represents the best action for the agent system.

3 Application: A Compromised Auction

The disclosure of bids to a compromised host can affect the intended, timely, outcome of an auction. A Vickrey auction is a sealed bid auction where the second highest bid is paid by the highest bidder [Vickrey, 1961]. All bidders maximize their payoff if they employ a truthful bidding strategy. Agents often use Vickrey auctions to acquire resources. Each agent submits its bid to an auctioneer host *hi* (the sink)—but unless the physical host of a bidding agent is directly connected to h_1, the message containing the bid must pass through other hosts in the agent system. All agents operating under normal conditions have neither the intent nor the capability of reading bids that are routed through their physical hosts.

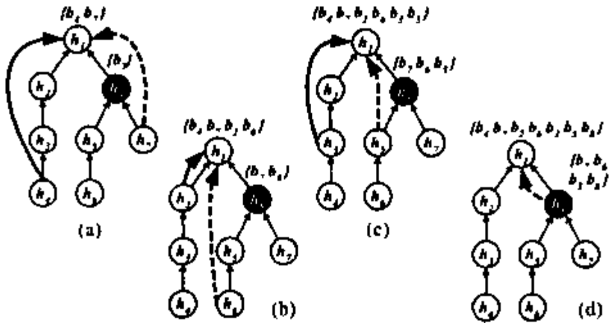

Figure 3: Bid propagation to host *h*\ in a Vickrey auction: (a) h_4 and h_7 bid (b) h_2 and h_6 bid (c) h_3 and h.$_5$ bid (d) h_8 bids

First, a compromised host, h_c can read all bids that are sent directly to or routed by means of h_c and potentially corrupt the auction. In this case, instead of maximizing absolute payoff, the bidding agents on h_c maximize their payoff *relative* to other bidding agents. In this type of "antisocial bidding" [Brandt & Weiss, 2001], assuming there are *n* bidding agents in the agent system, h_c most successful if it knows all bids 6i, b_2,..., b_n placed by *all* of the other bidding agents. The worst case is when all physical hosts use routes that contain h_c. In general this is not the case. Hence, if there are *n'* hosts that use routes containing h_c, the probability that the highest (or any) bid is disclosed to h_c is equal to n'/n.

Secondly, time is an issue: auctioneers are not willing to wait indefinitely for all bidders to respond. In any given decision problem there is some threshold, T, such that, if a bidder is more than τ hops from the auctioneer, its bid will not reach the auctioneer in time. Let U_i be the set of all hosts that communicate with host h_i via a route longer than r hops: $U_i = \{h_j : h_j \in H \wedge |R_{j,i}| > \tau\}$ U_i may contain hosts that use a route containing the compromised host. To adjust for this overlap, compute the set C_t of hosts affected by the compromised host, but not by the required message delivery time: $C_i = \{h_j : h_j \in H \wedge h_c \in R_{j,i} \wedge |R_{j,i}| \leq \tau\}$

operator	*m*	*t*	*V(N)*	*C1/n*	*U1/n*
ignore	0.286	0.250	0.509	0.5	0.0
reroute	0.714	0.321	0.598	0.25	0.25
remove	0.833	0.524	0.577	0.0	0.42857

Table 1: The terms and result *of V(N)* using α = 0.5, the probability of compromised messages $\frac{C_1}{n}$, and the probability of unreceived messages $\frac{U_1}{n}$ for the result of each operator.

The disclosure of bids to a compromised host during this decision procedure is illustrated in Figure 3. Assume for timing that *r* — 3. In this example, the probability $|C_1|/n$ is representative of the effect of compromised messages, and the probability $|U_1|/n$ is representative of the effect of time. As either probability increases, the value of the underlying mobile agent system should decrease.

Table 1 demonstrates how *V(N)* can be used to minimize the effect of a compromised host in a Vickrey auction. As the probability of compromised messages increases, the *message integrity rating* decreases. As the probability of unreceived messages increases, the time rating also increases. The operator yielding the highest value in this example is *reroute.*

4 Conclusions

This paper developed a utility-based model for agents to balance information assurance and network routing efficiency. We have discovered that there exists a natural tradeoff between information assurance and network routing efficiency for ad hoc mobile agent networks. Further, by empowering agents to decide for themselves how they communicate at the network level, one can increase the overall level of message integrity in an agent system. Our approach involves a novel exploitation of properties of ad hoc networks, enabling mobile agents to automatically adapt to changes that affect the security of their communication and migration. The capability to dynamically reason about the state of their network will provides new possibilities for secure computing.

References

[Brandt & Weiss, 2001] F. Brandt & G. Weiss. Antisocial agents and Vickrey auctions. *Wkshp on ATAL,* V 2333, pp 335-347,2001.

[Dix *et al.,* 2000] J. Dix, V. S. Subrahmanian, & G. Pick. Meta-Agent Programs. *1 Logic Prog.,* 46(1): 1-60, 2000.

[Forman & Zahorjan, 1994] G. H. Forman & J. Zahorjan. The challenges of mobile computing. *IEEE Comp.,* 27(4):38-47, Apr 1994.

[Royer & Toh, 1999] E. M. Royer & C. K. Toh. A review of current routing protocols for ad hoc mobile wireless networks. *IEEEPers. Comm.,* 6(2):46-55, Apr 1999.

[Vickrey, 1961] W. Vickrey. Counter speculation, auctions, and competitive sealed tenders. *J. Fin.,* 16(1):8—37, 1961.

[Xuan *et al.,* 2001] P. Xuan, V Lesser, & S. Zilberstein. Communication decisions in multi-agent cooperation: model and experiments. *AAAS,* pp 616-623, 2001.