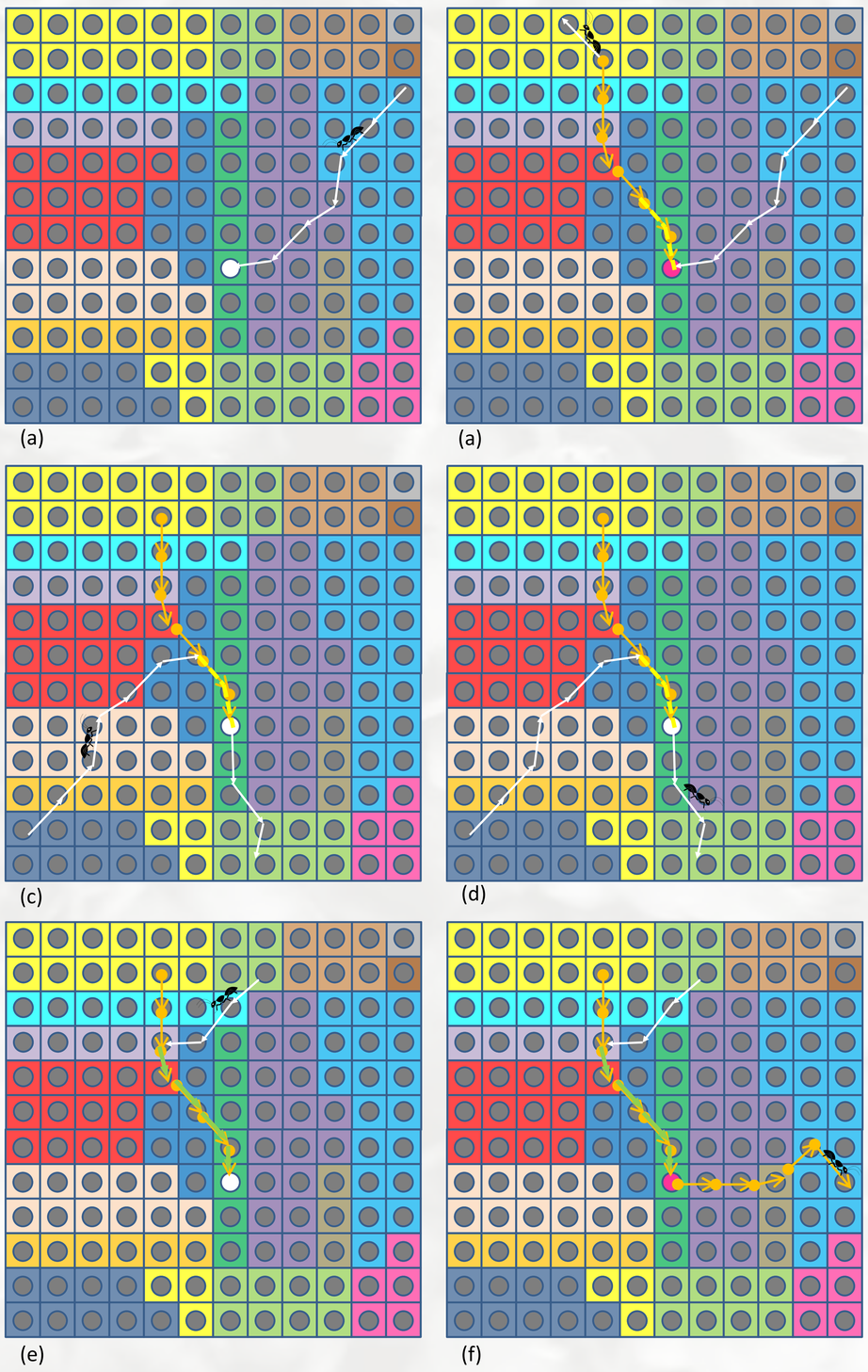
The Hive Mind: Applying a Distributed Security Sensor Network to GENI

Sean Peisert, University of California, Davis (PI); Matt Bishop, University of California, Davis;
Steven Templeton, University of California, Davis; Carrie Gates, CA Labs (CoPI)

Goal: Devise an effective, autonomous, decentralized security event monitoring system for GENI. It must operate in distributed, resource constrained environments without impacting operations. The system should be capable of detecting both local and distributed attacks, be itself secure, and operate with a minimum of human interaction.

Inspiration: Biological models are highly resource efficient and adaptable. Of particular interest are social behaviors where creatures communicate and interact. Social insects are a prime example. Ant foraging behavior is a lightweight decentralized method for resource discovery and exploitation based on communication using stigmergy (i.e. local modification of the environment as a means of communication). Other behaviors such as mobbing in crows, swarming in wasps, and immune systems use simple communication to induce group behavior, direct detection and response actions. Our primary motivation is that from a few simple rules and without external direction, complex behaviors can emerge. We investigate how these simple behaviors can be used to direct resources to support intrusion detection and reporting.

Basic Model



(a) A patrolling Mobile Sensor Agent (MSA) moves between nodes spanning multiple Aggregators and discovers activity matching its sensor function's target. (b) MSA moves away leaving a trail of "virtual pheromone" markers at nodes it passes through. These point toward the node where the discovered activity occurred. After a while it stops dropping and returns to patrolling. (c) Another patrolling MSA intersects a node with a marker and follows the trail to the discovered node. (d) No activity matches this MSA's sensor function. The MSA returns to patrolling. (e) Another MSA intersects the trail and follows the markers to the discovered node. (f) Activity on the node also matches this MSA's sensor function. The MSA moves off leaving its own trail.

As this behavior is repeated, many MSAs converge on the nodes in the vicinity. When the information gathered from the combined sensor functions, is sufficient to indicate reportable activity, alerts are sent to a supervising process where response may be initiated.

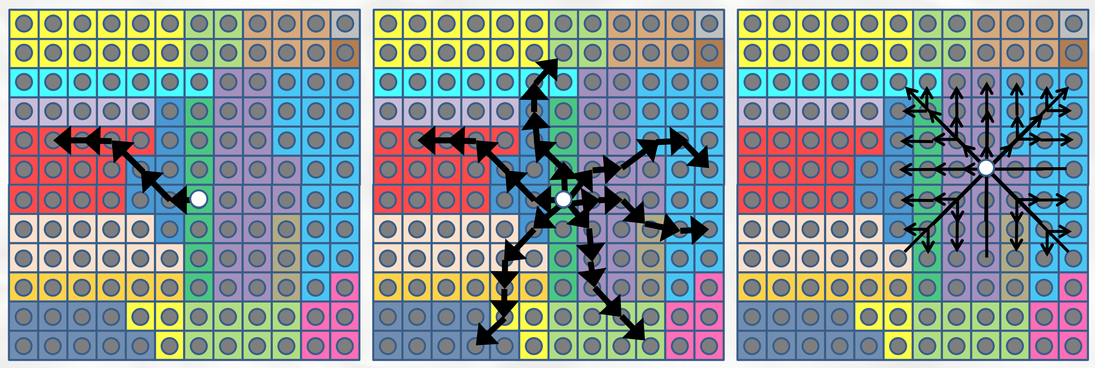
Challenges:

- Monitored nodes are distributed across multiple Aggregators. Each may enforce a different security policy.
- No central monitoring point.
- Monitoring method must not significantly affect Experiments.
- Distributed attacks can not be detected locally.
- Information about local attacks should be used to support detection of similar attacks on other hosts.
- Minimize amount of processing passed to supervising hosts.

Method: Devices in the system form nodes in a potentially dynamic mesh linking neighboring devices. This is shown in these figures as a grid, but arbitrary graphs can be used. Mobile Sensor Agents (MSAs), implemented as messages, move between nodes tending to follow a particular direction. The MSA's specific behavior depends on its state (patrolling, following, marking, etc.) and will generally execute an MSA specific sensor function on arrival. A process at each node executes the sensor functions, changes the MSA's state, maintains local state (e.g. is a marker here) and directs the motion of the sensors. MSAs carry minimal information: state, direction, age, and particular sensor type). Patrolling MSAs that find a node with interesting activity leave a marker trail to direct other MSAs toward that node. Patrolling MSAs encountering a marker trail will be directed along the trail toward the node where the interesting activity occurred. By directing many MSAs of different types (sensor functions) to the node, a picture of what is occurring can be created. Adjusting the creation/distruction rate of the MSAs, marker trail and dissipation rate allows us to tune the efficiency of the model.

MSAs can be either indicator of which predefined function to execute or be code-carrying mobile agents, or hybrids. The choice depends on the resource use requirements of the monitored system.

Alternative Communication Models



(a) *Basic model*: a single MSA leaves a trail from target in response to trigger. (b) *Alarm model*: multiple MSA are generated in response to trigger. (c) *Wasp model*: node Manager sends broadcast message to all neighbors in vicinity. Variations affect time to detect, resource use, and ability to detect non-local activity.

Detecting Distributed Attacks:

- The Basic Ant model alone is not sufficient. They are memoryless and do not directly communicate with each other. They can only detect local activity and direct other sensors to that vicinity. *Ants cannot detect global activity.*
- Neighboring activity must be obtained and monitored for trends.
- Additional communications methods and/or use of observed information by nodes allows detecting non-local activity.
- Mobile Sensor Agents can be extended to change behavior in response to behavior of others (e.g. swarm agitation)

Prototype: GENI experiment slices are swapped in with management a process running on each node. An extra node for security system oversight and reporting is added to the experiment. Experiments are being run to test a variety of performance criteria using Slices of up to 640 nodes. We are using ProtoGENI and DETER test beds and the Benito virtualization framework for our testing.

Project Webpage http://hivemind.cs.ucdavis.edu