Overview of Cooperative Infrastructure Defense and Ant Based Cyber Defense

Hayley Weiss
Hayley.Weiss@tufts.edu
Ming Chow

Abstract:

Today we monitor cyber defense by gathering data across an infrastructure to a single point and analyzing it centrally, which is problematic as it scales poorly. To combat this, the Pacific Northwest National Laboratory (PNNL) has been working on a method called Cooperative Infrastructure Defense (CID), which utilizes "digital ants" and "swarming intelligence" to quickly react and adapt to cyber attacks with humans supervising at the appropriate level. This paper gives an overview of the concepts behind CID and digital ants.

Introduction:

Sometime soon, ants might be crawling their way through your computer, but it isn't necessarily a bad thing. These ants aren't actual bugs, but are small digital models of ants that the Pacific Northwest National Laboratory (PNNL) started researching in 2006 as a means of detecting malware [4]. Using these Digital Ants and the eusocial behavior of ant colonies, PNNL is working on creating a Cooperative Infrastructure Defense, or CID, to create a more efficient and effective environment for combating security issues [4]. This method combines humans and machines in a way that allows humans to have the correct amount of control and interaction without letting them be a bottleneck for performance.

The Community:

When a cracker targets a program for attack, it becomes a cyber battle. The owners of the program are on the defense, and must attempt to locate and remove the vulnerabilities and hardware, while the attacker tries to continue the exploit as long as possible. A problem for the defense is that nowadays programs often span multiple organizations, which share a limited amount of communication and trust, making it very difficult to combat attacks. The defensive actions of one involved entity can even have detrimental affect on their partners [6]. Typically, to account for the involvement of multiple organizations all the cyber data is accumulated to a single point and analyzed centrally, but this approach scales poorly [3].

Current cyber-defense systems are multi-tiered with humans at multiple levels, but often too deep down for real effectiveness. This in-depth involvement require humans to make time-critical decisions, and in a crisis boundaries such as language, legal, proprietary, availability hinders information flow and makes communication slow and asynchronous [4]. Slow and static systems such as this aren't effective against attackers, who don't suffer from similar limitations and can move rapidly and concertedly [6]. And attackers are constantly coming up with slightly new ways to exploit flaws in programs. Current static models that are designed to defend against known threats wont catch new and creative exploits developed by attackers, even if they are only slight variations [1].

Cooperative Infrastructure Defense (CID) paired with Digital Ants, developed by PNNL, offers a possible complex dynamic system for defending against cyber attacks. CID uses a hierarchical framework mixing humans and machines. This design allows the system to rapidly adapt to new cyber attacks while letting humans supervise at the appropriate level [4]. This lessens the amount of necessary human involvement but still allows the ability for humans to intervene at will. The ants provide a flexible and simple method of identifying possible threats. Instead of looking for specific threats, the ants are only looking for symptoms of something being wrong, which can be hard to hide [5]. Digital ants also free up resources. Traditional anti-virus software scans on a set schedule and can take up a lot of computer resources. The numbers of ants rise and fall as problems are detected and only infringe on computer hardware when there is an actual threat to combat [2].

Digital Ants are still only in the study phase though and most likely will not be introduced into the public in the new few years. But PNNL has created simulations of the entire framework, as well as prototypes of the UI and the ants and created some proof-of-concept implementations [6]. Ant-Based Cyber Defense is a technology readiness level one project and the sentinels and sensors have been implemented on a 64 Linux virtual machine. With this implementation, ants were able to identify previously unknown malware that was based on real Linux worm code [3].

Action Items

Digital Ants

The inspiration for Digital Ants came from studying the actions of ant colonies in nature. While not particularly bright on their own, ants work together to build incredibly complex networks and armies and are capable of terrific feats.

The digital ants are incredibly basic. They aren't actually programs but are instead just messages that carry information about each specific ant. Since they are just messages, ants are implemented as IPv4 network packets. Both TCP and UDP protocols have been investigated and even though undetected packet loss can occur with UDP, if this drop rate is low then overhead of TCP can be avoided [6]. As ants are transmitted as single packets, the amount of information they hold is quite small, which limits the possibilities of what they can do.

Ant packets carry around 10 different fields pertaining to their unique identity. The names and properties of each of these fields can be seen in the table below [6].

Field	Description	Use
Id	Unique identifier for the ant.	Used to determine if a pheromone was left by itself
Sensor_type	The evidence type the ant is seeking	This tells the Sentinel what sensor function to execute
Sensor_parameters	Parameters for a particular sensor type	Allows for variants of the same sensor, e.g. thresholds, filenames, character sequences, etc.
State	Foraginging, following, dropping, idle	Determine an ant's actions.
Age	How long the ant has been traveling	After a period of time ants will die (i.e. be removed)
Direction	The direction vector for the ant	This is used to determine the next node for the ant when the ant is not following a pheromone trail.
Prior node	The host the ant was received from	Used to direct ants along pheromone trail.
Time_dropping	How long the ant has been dropping pheromone	After a period of time an ant will stop dropping and wander idle.
Time_idle	How long the ant has been idle	After a period of idle wandering ants will return to foraging.
Where_found	The location the evidence was found	Used in experiments for alternative ways for pheromone to direct ants to a target

The idea is that each individual ant is looking for something slightly different, with minds of having up to about 3,000 different kinds of ants all searching. Basically, the ants scatter in different directions across the network of machines, some looking for problems and others just wandering around. If an ant is old and hasn't found anything of interest it will die out. If it finds something interesting, it leaves behind a pheromone trail that results in positive feedback and leads to the creation of more ants of the same type and eventually a swarm of ants marking the potential problem [2].

A movement and pheromone model using pheromone vectors has been established in order to mimic the pheromone laying capabilities of actual ants. This model provides both greater stability for real world development and a directed random walk. A pheromone vector T is the sum of three vectors: D: the deposited pheromone B: the background pheromone and H: the heading bias pheromone such that T = D +B + T. The ant's next director, P, is found by normalizing T. The cumulative probability for each possible direction the ant can continue in is calculated, and a direction is randomly chosen from these probabilities.

Cooperative Infrastructure Defense

In CID, humans and various software agents share the responsibility of securing an infrastructure comprised of enclaves belonging to member organizations [3]. The hierarchy of CID consists of four main levels. From top of the chain to the bottom, these layers are supervisors, sergeants, sentinels and sensors.

Supervisors are at the top of the chain. This is the level where humans are involved. Since they are at the top they are able to supervise and guide the system if necessary. Supervisors can look over multiple enclaves, which are collections of machines owned by a single organization and managed under a single policy [3]. The supervisor interacts with the rest of the system through a graphical UI that gives situational awareness of the state of the enclave. The supervisor can adjust properties of the lower levels to influence behavior of the involved parameters [6]. When the lower-level agents encounter a situation that they deem to be problematic enough and require human involvement, they will contact the supervisor to take action [3].

The next level down is the sergeant, which is an enclave-level agent responsible for the security state of an entire enclave. They communicate with humans for guidance and to run the system as humans specified [3]. This communication is in the form of situation awareness delivered by an information visualization interface. They used supervised learning algorithms to become more efficient over time and are “heavy-weight” rational agents and thus make decisions based on logic. Using the supervisor’s specifications, they define the “geography” of the network for the enclave and broker agreements with other enclaves. During system initialization sergeants identify their sentinels and their neighbors. At runtime, they reply to sensor function definition requests and collect reports from their sentinels, the information from which they use to track alerts, ant motion and ant state. Eventually, the ability for sergeants to create new sensor types on their own will be implemented [6].

Then come the sentinels, which are at the host-level. They protect and configure each monitored machine and only interact with the human supervisors when they need clarification on classifying ambiguous evidence from their sensors [3]. The sentinel implements the policy they receive from the sergeant and interface with the sensors of the CID. Using a combination of evidence from the sensors, historical data and information from other sentinels, sergeants and supervisors the sentinel decides whether a problem exists and how to go about solving it. They provide local geography, mobility and provide rewards and spawning capabilities to the sensors. When they reward a visiting sensor, this will cause the sensor to deposit its digital pheromone trail to attract more sensors and the more sensors that visit the more information provided and the sooner a potential problem can be diagnosed [6].

There are a few main functions the sentinels are in charge of. I'll go through a few of these functions below. Because the pheromone trail left by the sensors isn't permanent, the sentinel is in charge of *evaporating the pheromone* by monitoring its age and removing them once a limit is reached. This prevents sensors from following trails based on data that is old and out of date. They have a *receive ant function*, that reads ant packets from the network and processes them sequentially. The *kill ant* function is in charge of removing ants by considering a combination of ant age and ant crowding. Once an ant reaches a certain age, it may die, and the likelihood increases with age. The same consideration goes for crowding. The *foraging* function occurs when the sentinel receives a "foraging" ant. It executes the sensor function appropriate for the ant's type and if it finds something interesting, it changes the ant's state to "dropping," else it will likely change the state to "following." Another function is *create ant*, which makes new ants if there is a low crowding factor and a high utility of a specific type of ant. The sentinel also has a series of *sensor functions,* which is determined by the type of the Ant the sentinel is interacting with. There are also the functions *following, idle, receive-sergent-message.*

The lowest levels of CID are the sensors, which are implemented as Digital Ants. Each ant uses a learning classifier to match a particular set of conditions in the machine they are visiting. There are two main categories of sensors: Markovian and differential. Markovian look for static conditions and differentials look for differences in conditions between hosts in their recent memory and the current host, which means they are looking for general indicators of something unusual happening with a machine [6]. They are looking at all parts of manner of things, such as connection rate and CPU utilization [2]. As mentioned earlier, the ants, and thus the sensors, are implemented as IPv4 network packets.

Conclusion

Cooperative Infrastructure Defense and Ant-Based Cyber Defense provide a promising method of combating security issues in large and complicated networks. By combining humans and machines into an efficient hierarchy, CID presents a dynamic security system that can move and adapt rapidly to new attack techniques, compared to the current static systems that are in place for computational products owned by multiple organizations. The social behavior of ants proves to be a good model for basing security approaches. While weak individually, ants can quickly join together to form an army that can overcome difficulties too big and unusual for one small entity. Hopefully some time soon, they'll be bugs in your system, and you'll actually want them there!

Works Cited

[1] “Ants vs. worms: New computer security mimics nature.” 25 Sep. 2009. [online] Available: http://phys.org/news173108776.html

[2] Bland, Eric. “Digital ‘ants’ take on computer worms.” 28 Oct. 2009. [online]. Available: http://www.nbcnews.com/id/33509921

[3] “DigitalAntsTM: Ant-Based Cyber Defense” Pacific Northwest National Laboratory. [online] Available: http://i4.pnnl.gov/news/digitalants.stm

[4] Fink, Glenn, Oehmen, Christopher. “Final Report for Bio-Inspired Approaches to Moving-Target Defense Strategies.” 30 Sept. 2012

[5] King, Anna. “New Model for Cyber Security: Ants.” 2 Aug. 2011. [online] Available: http://www.npr.org/templates/story/story.php?storyId=138941257

[6] J. Haack, G. Fink, et. Al. “Ant-Based Cyber Security” http://hivemind.cs.ucdavis.edu/_docs_/ITNG-290-Ant-Based-Cyber-Security.pdf