Bio-Inspired Cyber Security for Smart Grid Deployments

A. David McKinnon, Seth R. Thompson,
Ruslan A. Doroshchuk, Glenn A. Fink
Secure Cyber Systems Group
Pacific Northwest National Laboratory
Richland, WA, USA
david.mckinnon@pnnl.gov

Errin W. Fulp
Department of Computer Science
Wake Forest University
Winston-Salem, NC, USA
fulp@wfu.edu

***Abstract* — Smart Grid technologies are transforming the electric power system in ways that will significantly impact electric distribution systems and result in greater efficiency. However, the increased scale of the grid and the new types of information it will transmit introduce security risks that cannot be addressed by traditional, centralized security techniques. We propose a scalable approach inspired by the complex-adaptive control systems of social insects, such as ants and bees. These systems emerge from inter-agent communication and the collective application of simple rules. The Digital Ants framework is a bio-inspired framework that uses lightweight, mobile agents. The agents communicate using digital pheromones which enable the agents to alert each other of possible cyber security issues. All communication and coordination is both localized and decentralized thereby allowing the framework to scale across the large numbers of devices that will exist in the Smart Grid. Furthermore, being lightweight makes the agents suitable for implementation on devices with limited computational resources. This paper will provide a brief overview of the Digital Ants framework and then present results from testbed-based demonstrations that show how Digital Ants can identify a cyber security attack scenario against smart meter deployments.**

***Index Terms*-- Agents, Bio-inspired, Computer security, Cyber security, Smart Grid, Smart meters.**

I. INTRODUCTION

Modern energy delivery systems are typically large and complex, composed of intelligent control system devices with widely varying capabilities and real-time constraints. In the past, information flows have typically been one-way, from devices and meters in the field to control centers. Smart Grid devices and technologies will increasingly require two-way information flows. Additionally, the scale of managed devices will increase dramatically as smart meters and devices are deployed at customers' homes and businesses. As utilities build out their networks to accommodate both scalability and bi-directional information flows, they need to address new cyber security challenges. These challenges will bear a resemblance to traditional cyber security challenges faced by corporate information technology (IT) and communication networks, but they will also have some striking differences that stem from the nature of these control systems. Thus, traditional tools (e.g., intrusion detection systems) and best practices used today for securing corporate environments may fail when applied to the energy delivery systems.

The Digital Ants biology-inspired solution presented in this paper is lightweight and scalable—suitable for deployment in energy delivery systems. This solution employs the Digital Ants Framework (DAF), which applies lessons learned from ant foraging behaviors to distributed cyber security problems. The DAF utilizes lightweight software agents that use stigmergic (pheromone-based) communications to create useful emergent colony behaviors that ensure the security of the protected enclave. The application of the DAF to energy delivery systems incorporates data from both information technology and energy delivery systems. This framework and a set of experimental results are described in the following sections: Section II presents related work, Section III introduces the Digital Ants Framework and provides details on its current implementation; Section IV is devoted to a plausible cyber security scenario that was used during the evaluation, the results of which are presented in Section V; after which the paper concludes in Section VI.

II. RELATED WORK

Mobile agents can offer a robust and distributed method for managing the operation and security of critical infrastructures such as the Smart Grid [1]. Their application has been varied; for example, agents have been proposed to identify system disturbances [2], and provide power flow situational awareness [3] and voltage control [4]. There has also been extensive research in the application of agents to control the operation of micro-grids [5], [6], [7]. The use of Digital Ants, as proposed in this paper, is unique in that it leverages mobile agents and a form of swarm intelligence to provide cyber security.

This research was supported, in part, by the U.S. Department of Energy under U. S. Department of Energy Contract DEAC05-76RL01830.

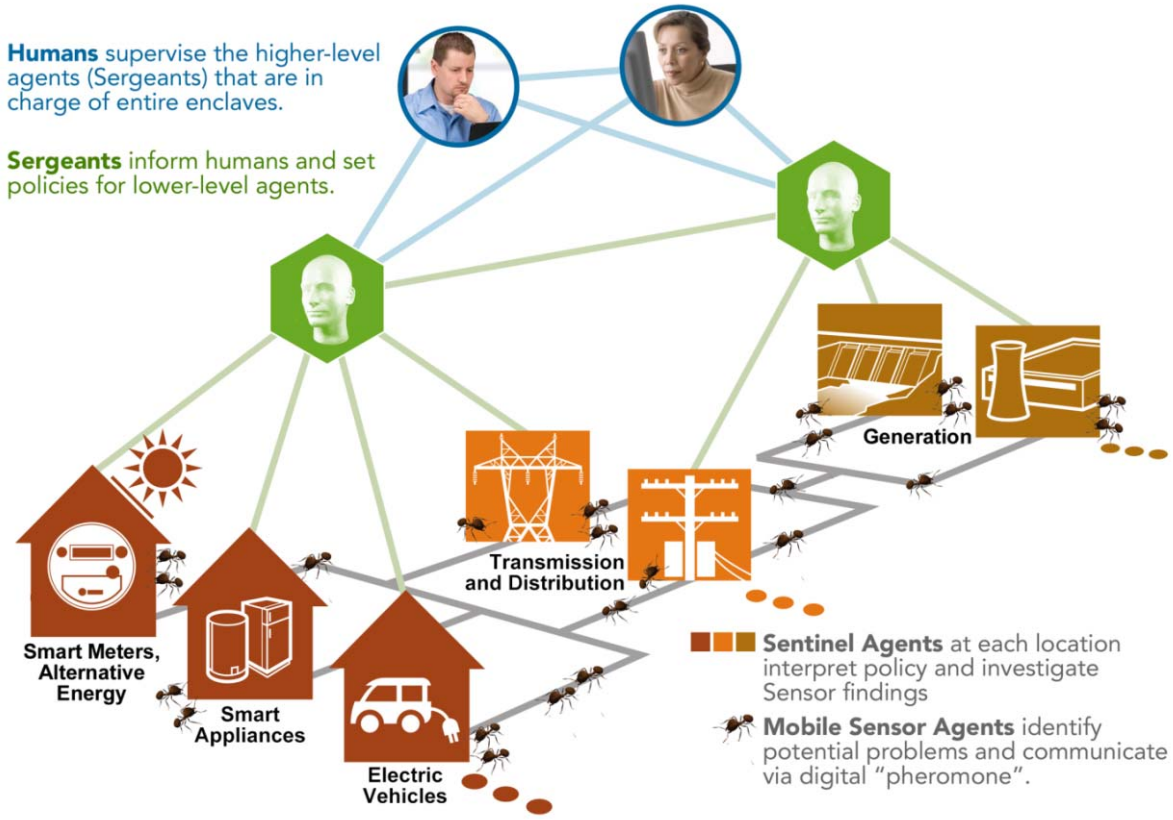

Figure 1: Digital Ants hierarchy

III. Digital Ants Framework

The DAF uses lightweight sensor agents that roam a protected network(s). Additional layers in the framework's hierarchy enable a mixed-initiative approach [8] with oversight and influence from a human operator. Each of the framework's key components, the Sensors, Sentinels, Sergeants, and Supervisor, will be described in the following subsections. A visual overview of the framework is shown in Figure 1.

A. Sensors

At the bottom of the Digital Ants framework hierarchy are the Sensors—mobile, lightweight software agents. These Sensors roam from device to device, observing one specific type of information or state (e.g., CPU usage, network bandwidth, or power consumption) on the monitored device. Sensors compare each device's measurement to values from neighboring devices or to a critical threshold in order to discover anomalies. The Sentinel must then determine whether the Sensor's finding is normal for that specific device or whether further investigation should be undertaken. If the latter, the Sentinel rewards the Sensor and the Sensor departs, leaving behind a trail of messages that act like the digital equivalent of pheromones. This trail attracts other Sensors that are looking for other types of anomalies, which if found, will cause more pheromones to be dropped. This creates a positive feedback loop that quickly attracts a swarm of Sensors, each of which will observe a different behavior at the location of concern. In short order, a holistic (multi-parameter) view of the device's behavior will emerge.

B. Sentinels

Sentinels are stationary agents that reside at a given device. Visiting Sensors provide the Sentinel with data about conditions at its devices as well as at neighboring devices. These data are used to develop a notion of normal and to identify anomalous behavior. Significant events are communicated up the hierarchy to the Sergeant.

C. Sergeants

Sergeants are responsible for large enclaves of similar (either by function or location) devices. Since multiple Sentinels report to a single Sergeant, each Sergeant is able to correlate and filter information before presenting it to a human supervisor. Without this step, humans could be overwhelmed with the velocity and magnitude of data produced by the Sensors. Furthermore, Sergeants can potentially discover distributed threats that emerge across multiple Sentinels. Sergeants also have the role of adapting human-level policy directives into actionable commands and configuration changes that can be executed by lower level agents and devices within the framework.

D. Supervisor

A key premise of the Digital Ants framework is that operators should be in the "right" loop. Ultimately, operators are responsible for the operation of energy delivery systems.

When operators are deluged with overwhelming amounts of data, they can no longer focus on issues within the system. When it comes to cyber security this is a significant loss because operators must be continually vigilant with respect to zero-day attacks.

E. Smart Grid Application

Digital Ants map well to energy delivery systems, perhaps even better than to typical IT systems. Energy systems are organized and controlled hierarchically and separated into distinct units with well-defined divisions of function. This organization can easily be used to create the enclave boundaries that Digital Ants will use. Whether considering smart meters, pole-top devices, or smart sensors and actuators in a generation facility, the number of devices on which Digital Ants will need to operate is very large—tens of thousands of devices will be common. This provides a great variety and volume of information that the Digital Ants' decentralized approach will scale to meet. Furthermore, because it is based upon a lightweight agent framework, even small devices such as smart meters can be augmented with the required Sensor and Sentinel logic. The Digital Ants Framework was designed to use a variety of sensor types, which means that the framework will process alerts generated from Smart Grid observables (e.g., meter disconnects) side-by-side with alerts derived from the observation of traditional IT operations (e.g., network bandwidth, CPU usage).

IV. SMART METER ATTACK SCENARIO

Our current research is focused on adapting the Digital Ants Framework to energy delivery systems. To keep our development priorities and evaluation strategy relevant to the Smart Grid, we developed a plausible smart meter attack scenario. Previous work on the Digital Ants Framework focused either on general purpose agent movement and pheromone studies [9], [10], or used a naïve attack model based upon the spread of a simple worm [11].

Our scenario must accommodate the widely varying scale of smart meter deployments, from a few thousand in small utilities to millions in large utilities. Furthermore, given that smart meters are resource-limited devices our scenario must emphasize the need for a very lightweight framework.

A. Attack Scenario Overview

Assume a group wishes to extort a utility by threatening to blackout a large portion of the utility's service area. The blackout will be carried out through the control of a large number of compromised smart meters. The utility's smart meters will be compromised via a worm that, once in place, will allow the attacker to remotely disconnect power to the customer site. Furthermore, assume that the worm is engineered to report normal power consumption after a malicious remote disconnect command is issued in order to partially obscure the group's actions. Before the mass disconnect is executed, the group will randomly disconnect a small number of meters in a test phase that will prove to the utility that they have the technical ability to carry out their extortion threat.

B. Attack Scenario Implementation

Our plausible scenario is a hybrid attack that has both traditional computer/network attack signatures and power systems impact. Ideally, a hybrid testbed with both traditional IT features as well as power systems capabilities would be used. Unfortunately, hybrid testbeds that can be used for utility-scale deployments do not exist. Therefore, we chose to develop two sub-variants of our scenario, each of which was focused on a given aspect of the scenario (i.e., information technology and power systems) and an associated testbed suited for testing that sub-variants' focus. The first sub-variant focused on traditional IT metrics and used the DETER testbed [12]. Sensors developed for this testbed focused on identifying the malware via its standard IT-based observables (e.g., CPU usage, network bandwidth) as the worm propagates from meter to meter. The second sub-variant had a power systems focus and was implemented within the GridLAB-D environment where researchers have access to realistic power flow simulations. Thus its sensors have a power systems/energy delivery system focus (e.g., voltages, currents, power flows). In essence, we assumed that the smarter meter worm had already successfully spread undetected and therefore only the impact to the utility power system from the remote disconnects would be observed by the Digital Ants Framework.

V. EXPERIMENT

We conducted experiments in two testbed platforms: DETER (focusing on traditional IT metrics), and GridLAB-D (focusing on power systems-related measures). The current Digital Ants framework implements Sensors, a simplified Sentinel, and a minimalistic Sergeant. We developed multiple sensor types for both our target testbeds as well as additional small, embedded devices.

A. DETER Experiment

DETER is a highly reconfigurable testbed of roughly 200 physical nodes, operated by UC-ISI[12], targeted towards cyber security research. The testbed is isolated so that real malware can be released without the risk of infecting external nodes. A DETER experiment is defined through configuration files in which researchers assign experimental parameters including: links between nodes, bandwidths, packet loss, etc.; as well as the operating system and any software or scripts to execute. Because DETER uses physical rather than virtual nodes it provides a high level of fidelity to the sensor readings; CPU, RAM, disk, and network statistics all reflect values read from physical devices rather than simulated values.

The experiments conducted in DETER were designed to emulate a residential subdivision of 64 smart meters interconnected in a mesh network. The network topology was defined as a toroidal overlay, providing a consistent graph environment for Sensors to wander [13]. Because DETER provides four physical network cards for each machine, we simply connected each node directly to four neighbors, thereby creating the desired toroidal mesh for the Sensors.

The Sensors were tested against a piece of real malware—a variation of the Linux-based Cinik worm—modified to infect the experimental nodes. Once a node has been infected,

the worm scans for other suitable nodes to infect (both the internal and external address space, as done in the original worm). In addition, the worm opens a network port for command and control messages from the attacker. Testing against an actual worm, even if it wasn't a smart meter worm, was sufficient for this experiment to meet its IT-focused objectives.

Two different classes of Sensors were implemented: threshold monitoring Sensors and differential detection Sensors. The threshold Sensors monitor for values that are outside a predefined set of bounds. For example, a CPU utilization threshold Sensor may be configured such that any CPU reading above 75% triggers an alarm. A differential Sensor compares its history of readings to the current value and attempts to identify any anomalies. For example, a network connection Sensor with a history of [6, 7, 5, 8, 5, 6, 6] may trigger an alarm if a value of 12 is read because it is an outlier. Our differential Sensors maintain a memory of the last 30 readings and use this information to perform both the CUSUM algorithm [14] for change point detection as well as a calculation of means and standard deviations. Any anomaly in the data causes an alarm to be raised. The DETER experiments used 11 Sensors: CPU, firmware checksum, memory usage, network connection counts, network bandwidth, network entropy (ingress and egress) [15], and disk IO; where CPU, memory, and network connection count were implemented as both threshold and differential variants.

Most of the DETER Sensors function as their name implies, therefore we will only describe of a few Sensors in detail. The network entropy Sensors were implemented in order to detect an unusually large number of connections to and from widely varying addresses, which might indicate scanning behavior. A large number of connections is also detectable by the network connection threshold Sensor which monitors the total number of established network connections. The threshold is set at a typical value. If the node's behavior deviates significantly from this value, the threshold alarm is triggered. Finally, the firmware Sensor is a special class of threshold Sensor that monitors changes in the hash of specific system files manipulated by the worm. Any change in a file's hash constitutes an alarm.

Three experiments were performed in which the worm was released on one of the 64 vulnerable nodes (smart meters). The experiments were allowed to run for up to 30 minutes, or until all nodes were infected. The Sensor performance was collected for each experiment, specifically which groups of Sensors detected the worm and which Sensors commonly participated in the detection. The tested Sentinel implementation required three unique Sensors to alarm before reporting an infection to the Sergeant. These three Sensors are referred to as a Sensor alert triplet. The most common alert triplets used to report infections are shown in Figure 2.

An important feature to note is that more than 15 unique triplets were significant enough to trigger reporting infections to the Sergeant. The diversity of alerts is strong support for our claims of adaptability and resilience; it also shows that the framework does not rely on specific signatures to detect infections.

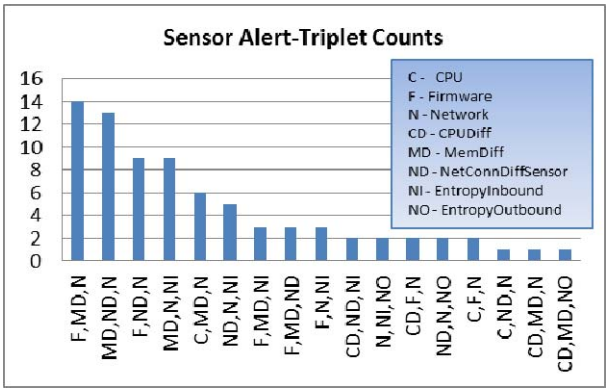

Figure 2: Common triplets detected together by sensor type

Interestingly, three of the Sensors did not generate alerts: the network bandwidth, disk usage, and memory threshold Sensors were not observed in any alert triplet or in the total Sensor counts (see Figure 3). This is because the worm did not require large amounts of bandwidth per scan action. In addition, the worm requires little memory to operate and does not write to the disk. Therefore, the worm operated below those threshold values and did not trigger any alarms. The most useful Sensors were network threshold, differential memory, firmware, and differential network connections; these Sensors accounted for nearly two-thirds of the alarms triggered.

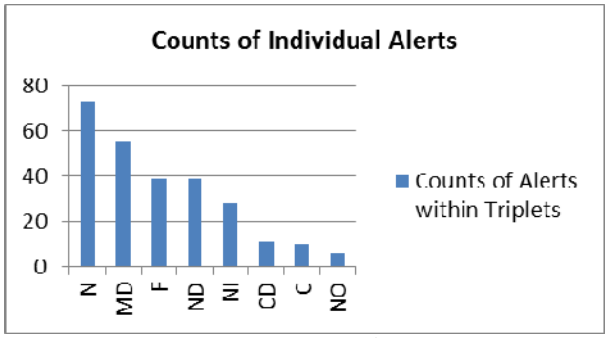

Figure 3: Individual alert frequencies

These experiments highlight the resiliency of the Digital Ants solution. By relying on a diverse set of Sensors, the system is able to adapt and respond to threats even in the absence of specific readings; a crucial capability for Smart Grid applications where Sensors or Sentinels may be attacked at any time.

B. GridLAB-D Experiment

The other set of experiments was conducted in GridLAB-D. GridLAB-D is a software tool that simulates large-scale power distribution systems by solving complex sets of differential equations [16]. We leveraged the output of this tool by creating Sensors that were able to parse and process the generated log files containing the steady-state power flow information.

Unlike the DETER experiments where a consistent topology was used for all experiments, in the GridLAB-D [16] power distribution focused experiments, multiple topologies were used. The first topology used was the standard IEEE 13 node model [17], containing 19 residential transformers. The rest of the topologies used were generated from models of real distribution systems in various regions of the U.S. created by the GridLAB-D team. For each topology, one to five residential houses were connected to a residential transformer. The number of transformers was in the order of hundreds, depending upon the specific topology. GridLAB-D was used to generate the power flows at the house and transformer levels. GridLAB-D was not designed with the goal of supporting cyber security testing, therefore in order to support our experiments we had to create two sets of power flows: normal and malicious. Infected smart meters reported values from the normal flows, but the transformers reported values from the malicious flows that had remotely disconnected meters. This approach enabled the misreporting of power flow data that was required by our attack scenario

Three main Sensors were used in the GridLAB-D experiments: a power flow mismatch Sensor, an uptime Sensor, and an uptime delta Sensor. The power flow mismatch Sensor compares the power flow readings at the residential transformer and residential smart meter levels. The uptime Sensor measures the elapsed time from a remote disconnect. The uptime delta Sensor measures the time between the last two remote disconnects.

In our scenario, we designed smart meter malware behaviors to mask remote disconnect outages by reporting normal power flow readings rather than zero power consumption. The power flow Sensor aggregated the power flows from all of the houses attached to a transformer and compared this with the transformer power flow values. If the readings were too far apart, allowing for losses, the Sensor would report a problem to the Sentinel. This would be an indication that one or more smart meters were potentially falsifying power flow values.

The scenarios assumed that each residential meter has a remote disconnect ability, allowing the utility to disconnect the house from the power system without dispatching a technician. Furthermore, we assumed that each smart meter keeps track of the time between the last two disconnects. A remote disconnect might be required if a customer moves out or neglects to pay the bill. Once the bill is paid or a new resident moves in, it is unlikely that another disconnect will occur in the near future. The uptime delta Sensor flagged any houses having multiple remote disconnects within a short time period. The uptime Sensor simply reported a problem when a single remote disconnect had occurred recently.

The Sentinel logic works as follows. Reports from uptime delta Sensors were treated as attack detections since remote disconnects are rare in normal cases. These alerts would indicate exactly which houses had been disconnected. Reports from power flow mismatch Sensors and uptime Sensors were treated as evidence of an attack; power flow mismatches and low uptimes alone may occur naturally, but in combination, it is likely that an attack has occurred. Therefore, if reports from both of these Sensors were received at a Sentinel, it was also considered a detection of an attack. These detections would indicate that one of the houses connected to the transformer in question had been attacked. However, because the alerts are triggered at the transformer level, the exact location of the attack is not found—it is only narrowed down to one of five houses.

The smart meter attack scenario specified a three-stage attack. The DETER-based experiments focused on the first stage, namely worm propagation. The GridLAB-D experiments focused on the middle and final stages of the attack scenario. In the middle stage, a small subset of houses experienced random, intermittent outages. These outages were followed by larger scale, synchronized outages in the final stage of the attack. During our GridLAB-D experiment, outages during the middle stage caused uptime Sensors to report problems to many of the Sentinels. Reports from the power flow mismatch Sensors enabled identification of the transformers that supplied power to the houses under attack. During the final stage of the attack, the uptime delta Sensors identified houses that had experienced outages in both the middle and final stages.

Results from the GridLAB-D experiment are shown in Figure 4. From day 1 to day 3.5, the middle stage of the scenario, the test stage, was executed as indicated by the slow and steady increase in infection reports. At day 3.5, the final stage of the scenario begins and a coordinated disconnect of 40% of the meters is executed. This is reflected in the graph by the significant increase in infection reports. Shortly after day 4, the meters are reconnected to the grid, explaining why the graph then levels out until the end of the experiment.

Our experiments with GridLAB-D showed how Digital Ants can be used to detect faults and attacks in electric grids in addition to how they have been shown to be effective in standard IT scenarios.

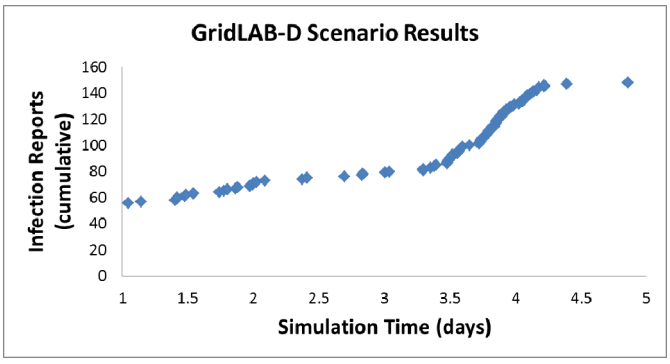

Figure 4: Cumulative infection reports

C. Discussion

We successfully conducted two separate experiments that addressed all three stages of our smart meter attack scenario. The DETER experiment demonstrated the resilient nature of Digital Ants' Sensors. Granted, an intrusion detection system configured with an appropriate signature could have easily

identified the Cinik worm. But identification was not the point of these experiments. Rather the experiments demonstrated that simple Sentinel logic could leverage decentralized data from multiple and varied Sensors, without *a priori* knowledge of a given attack. The GridLAB-D experiment demonstrated that the Digital Ants framework can use sensors based upon non-IT centric metrics (e.g., power system metrics). Utilizing Sensors from multiple domains will aid the defender. For example, even if one assumes that a stealthy smart meter worm can spread without detection, the GridLAB-D experiment demonstrated that Digital Ants Framework successfully used power system observables to identify this paper's cyber-physical attack scenario.

VI. CONCLUSIONS

The Smart Grid will introduce a large number of intercommunicating devices. Unfortunately the scale of the Smart Grid, both in terms of the number of devices and communications breadth, will render traditional, centralized cyber security approaches ineffective. Therefore, the Smart Grid will need highly decentralized techniques to secure this vital infrastructure.

We demonstrated the ability of Digital Ants, a new bio-inspired technology, to enhance cyber security within energy delivery systems. By demonstrating Digital Ants on real-world hardware in DETER we showed that our framework works well for IT systems. By demonstrating Digital Ants' utility in a high-fidelity electric grid simulation in the GridLAB-D environment, we showed its relevance to power systems and the Smart Grid in particular. Together these experiments show that the framework is effective. Our experiments provided evidence that Digital Ants technology is a decentralized solution that has the potential to scale to the levels needed in real-world deployments. Furthermore, by conducting simulations in multiple environments, we showed that Digital Ants technology can identify both power and IT/communications anomalies and cyber threats that impact energy delivery systems. All of these concrete results provide a path for industry acceptance and commercialization.

Future research is needed to identify, implement, and quantify the resilience of new Digital Ants Sensors and Sentinel logic. Demonstrating the Digital Ants framework with multiple malware variants and additional attack scenarios is also needed. We will also continue to increase the scale of the experiments.

VII. ACKNOWLEDGEMENTS

The authors thank Jereme Haack, Keith Fligg, Art McBain, Kyle Monson, and Scott Cooley for their contributions to the Digital Ants Framework.

This research was supported, in part, by the U.S. Department of Energy under U. S. Department of Energy Contract DEAC05-76RL01830.

REFERENCES

[1] R. Roche, B. Blunier, A. Miraoui; V. Hilaire, A. Koukam, "Multi-agent systems for grid energy management: A short review," In Proceedings of the 36th Annual Conference on IEEE Industrial Electronics Society (IECON 2010), pp.3341-3346, Nov. 2010

[2] T. Nagata and H. Sasaki, "A multi-agent approach to power system restoration", IEEE Transactions on Power Systems, May 2002, Vol. 17, pp. 457-462.

[3] H.F. Wang, "Multi-agent co-ordination for the secondary voltage control in power system contingencies", In *Proceedings of IEE Generation, Transmission and Distribution*, Jan 2001, Vol. 148, pp. 61-66.

[4] L. Cristaldi, A. Monti, R. Ottoboni and F. Ponci, "Multi-agent based power systems monitoring platform: a prototype", In *Proceedings of IEEE Power Tech Conference*, June 2003, Vol. 2, 5pp.

[5] A. Dimeas and N.D Hatziargyriou, "A multi-agent system for microgrids", In *Proceedings 2004 IEEE Power Engineering Society General Meeting*, Vol. 1, pp. 55-58.

[6] K. Butler-Purry, N. Sarma and I. Hicks, "Service restoration in naval shipboard power systems"; In *Proceedings 2004 IEE Generation, Transmission, and Distribution*, Vol. 151, No. 1.

[7] M. Pipattanasomporn, H. Feroze, and S. Rahman, "Multi-Agent Systems in a Distributed Smart Grid: Design and Implementation" In *Proceedings of IEEE PES Power Systems Conference and Exposition* (PSCE'09), Mar 2009.

[8] M.A. Hearst, 1999. Trends and controversies: mixed-initiative interaction. IEEE Intelligent Systems, 14 (5) (1999), pp. 14‑23

[9] Crouse, M.B.; White, J.L.; Fulp, E.W.; Berenhaut, K.S.; Fink, G.A.; Haack, J. "Using swarming agents for scalable security in large network environments" in the Proceedings of the IEEE 54th International Midwest Symposium on Circuits and Systems (MWSCAS), 2011.

[10] Fink, G., Berenhaut, K. and Oehmen, C. (2012) Directional Bias and Pheromone for Discovery and Coverage on Networks, to appear in *Proceedings of the Sixth IEEE International Conference on Self-Adaptive and Self-Organizing Systems*, Lyon, France; 10-14 September 2012.

[11] Haack, J.N.; Fink, G.A.; Maiden, W.M.; McKinnon, A.D.; Templeton, S.J.; Fulp, E.W. "Ant-based Cyber Security" in *Proceedings of the 8th Information Technology: New Generations* (ITNG 2011), pp. 918-926, 2011.

[12] The DETER Project. [Online]. Available: http://deter-project.org/.

[13] Fulp EW, M Crouse, and AD McKinnon. 2011. "Using Swarming Agents for Smart Grid Security," in *Proc. of the 7th Annual Cyber Security and Information Intelligence Research Workshop*, Oak Ridge National Laboratory, TN. PNNL-SA-82899.

[14] Guanhua Yan, Zhen Xiao, and Stephan Eidenbenz. 2008. Catching instant messaging worms with change-point detection techniques. *In Proceedings of the 1st Usenix Workshop on Large-Scale Exploits and Emergent Threats (LEET'08)*, Fabian Monrose (Ed.). USENIX Association, Berkeley, CA, USA, Article 6 , 10 pages.

[15] Ashwin Lall, Vyas Sekar, Mitsunori Ogihara, Jun Xu, and Hui Zhang. 2006. Data streaming algorithms for estimating entropy of network traffic. *In Proceedings of the joint international conference on Measurement and modeling of computer systems (SIGMETRICS '06/Performance '06).* ACM, New York, NY, USA, 145-156. DOI=10.1145/1140277.1140295 http://doi.acm.org/10.1145/1140277.1140295

[16] GridLAB-D, [Online]. Available: http://www.gridlabd.org/

[17] IEEE, Radial Distribution Test Feeders. [Online]. Available: http://ewh.ieee.org/soc/pes/dsacom/testfeeders.html.