Hive Oversight for Network Intrusion Early Warning Using DIAMoND: A Bee-Inspired Method for Fully Distributed Cyber Defense

Maciej Korczyński, Ali Hamieh, Jun Ho Huh, Henrik Holm, S. Raj Rajagopalan, and Nina H. Fefferman

The authors investigate the potential for a self-organizing anomaly detection system inspired by those observed naturally in colonies of honey bees. They provide a summary of findings from a recently presented algorithm for a nonparametric, fully distributed coordination framework that translates the biological success of these methods into analogous operations for use in cyber defense and discuss the features that inspired this translation.

ABSTRACT

Social insect colonies have survived over evolutionary time in part due to the success of their collaborative methods: using local information and distributed decision making algorithms to detect and exploit critical resources in their environment. These methods have the unusual and useful ability to detect anomalies rapidly, with very little memory, and using only very local information. Our research investigates the potential for a self-organizing anomaly detection system inspired by those observed naturally in colonies of honey bees. We provide a summary of findings from a recently presented algorithm for a nonparametric, fully distributed coordination framework that translates the biological success of these methods into analogous operations for use in cyber defense and discuss the features that inspired this translation. We explore the impacts on detection performance of the defined range of distributed communication for each node and of involving only a small percentage of total nodes in the network in the distributed detection communication. We evaluate our algorithm using a software-based testing implementation, and demonstrate up to 20 percent improvement in detection capability over parallel isolated anomaly detectors.

INTRODUCTION

Over the past years, cyber-attackers have taken advantage of the massive acceleration in the adoption of virtualization and cloud computing, the Internet of Things (IoT), and mobile devices as an increase in potential targets and expanding attack surface. Motivations are the major characteristics that differentiate malicious actors. Organized crime is interested in economic gain, nation-states are mostly interested in cyber-espionage, whereas hacktivists can be motivated politically or ideologically.[1] Cyber-attack strategies have also evolved significantly: modern malicious activities are spread stealthily over a large number of malicious machines. Those can be compromised or rented from so-called bulletproof hosting providers that ignore all abuse notifications [1]. This increases the chance of cyber-criminal success, either decreasing the probability the attack will be noticed or launching a distributed denial of service (DDoS) attack as a smokescreen to cover virus or malware installation, and/or financial or data theft.[2]

To address these more challenging types of cyber-attacks, recent defenses have introduced the idea of sharing information across organizational boundaries, allowing collaboration to achieve rapid detection and mitigation for a variety of cyber-attacks, especially those for which prior knowledge is scant or nonexistent. Indeed, an entire new infrastructure is being created with new sharing protocols, cyber threat "exchanges," and government backing. Automated cyber data processing and sharing is already being promoted as the new defensive strategy against smart and highly distributed adversaries. However, there are some fundamental challenges to address before this paradigm can become reality, such as:

- Policy issues that prevent sensitive data from being shared between organizations
- System scalability
- Semantics of the data being exchanged
- Alert correlation

As cyber-attacks are evolving rapidly, the data captured in one particular environment may be incomparable to data from another, vitiating any gains from sharing. Any form of detection that relies on comparison of semantically rich data is thus in jeopardy if the data comes from sensors in different domains. Even if direct comparison is possible, it is not guaranteed that the existing alert correlation techniques will be able to reconstruct novel, complex attack scenarios.

HONEY BEES AS AN EVOLVED ANOMALY DETECTION MACHINE

Colonies of honey bees rely on foraging workers to discover and share locations of flowering plants from which to gather the pollen and nectar used for food. The colony operates under many time-varying constraints: different plants flower at different times of year and/or day, other ani-

[1] http://www.mcafee.com/mx/resources/reports/rp-quarterly-threats-aug-2015.pdf

[2] https://www.neustar.biz/resources/whitepapers/ddos-protection/2014-annual-ddos-attacks-and-impact-report.pdf

Maciej Korczyński is with Delft University of Technology; Ali Hamieh and Nina H. Fefferman are with Rutgers University; Jun Ho Huh and S. Raj Rajagopalan are with Honeywell ACS Labs; Henrik Holm is with Forest Glen Research, LLC.

0163-6804/16/$25.00 © 2016 IEEE

mals also eat the plants/flowers, or the nectar and pollen are depleted by both direct competition with other insects/bees and by their own colony mates having already gathered the resources, making additional trips redundant. Each of these challenges must be met efficiently since the rate of resource acquisition determines the probability of colony growth, reproduction, and survival through the winter [2]. Meeting these challenges requires the colony (using only the relatively simple cognition and communication available to bees) to identify locations richest in resources, communicate their location to comrades, exploit them quickly, and abandon depleted locations rapidly in favor of alternate sources. Honey bees manage to meet these challenges with startling efficiency by a very simple method: each forager evaluates each site they visit; if a forager is excited by the resource richness of the site, she returns and tells a subset of her comrades the location of the resource and her own relative level of excitement (via mathematical dance language). Bees who receive her signal decides whether or not she was excited enough to merit their own trip to the site. If they go, they either return just as excited to recruit others, or else disagree, decide the site was not exciting enough, and search for a new site themselves or wait for another comrade to recruit. This system fulfills many desirable features: excitement waxes and wanes endogenously with site quality, sites are exploited while also searching for new sites, individuals identify new sites that do not fit the current predominant interest, and attention accrues very rapidly at any site consensus deems worthwhile without the need for bees to agree a priori on any single definition of "exciting."

Putting Bees to Work in Cyber-Defense

In this article we define HONIED: Hive Oversight for Network Intrusion Early Warning using DIAMoND — a bee-inspired method for fully distributed cyber defense. Our research is the first to investigate the potential for a self-organizing anomaly detection system inspired by the distributed algorithms colonies of honey bees use to forage efficiently to provide appropriate, dynamic detection thresholds for anomalous event patterns on computer system networks to improve early detection and mitigation methods to counter malicious threats.

Our approach addresses some of the main challenges of distributed defense strategies. The proposed system allows for cooperation between sensors in an arbitrary virtual topology and does not rely on sharing the particulars of the underlying event, but only the pattern of "excitation" seen in the sensors. By its nature this data does not contain any individually sensitive information, or even any information about the specific attack. We expect that overcoming organizational hurdles that may prevent sharing of such data would be far easier. For the same reason, our scheme easily addresses the third and fourth challenges; because the data shared is very simple (not even individual values for detection thresholds are shared), there is no question of creating semantic equivalence or complex correlation techniques. Finally, the scheme enables sensors to self-tune their individual detection threshold values using a feedback mechanism. When new attack patterns appear, the sensors learn by cooperation to sense them — it takes some time, but there is no prior modeling that has to be applied to the sensors. That makes our scheme especially appealing for detecting novel network attacks assuming that some controls (e.g., local intrusion detection systems) are able to detect their symptoms.

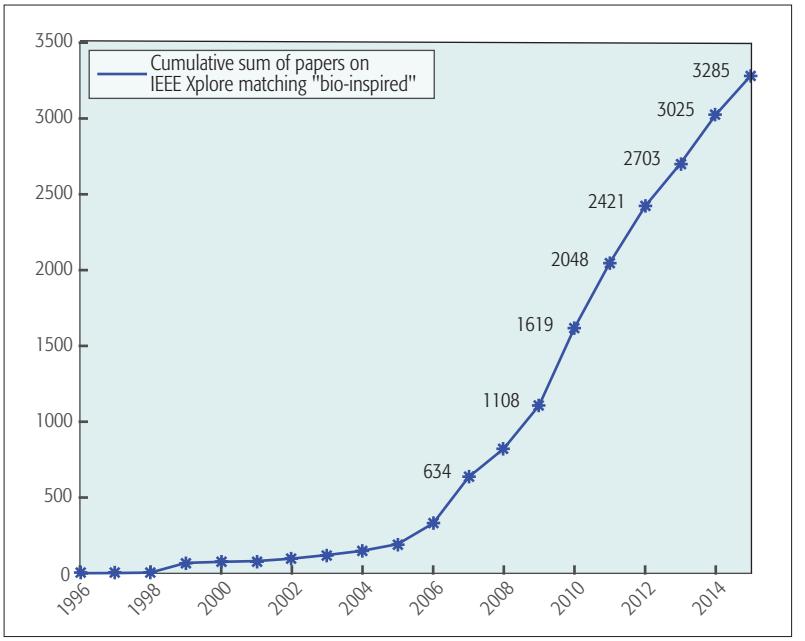

Figure 1. Literature in bio-inspired algorithms.

Related Work

There have been several proposals for fully distributed systems [3–7]. Locasto *et al.* proposed a fully distributed peer-to-peer (P2P) intrusion detection system (IDS) called Worminator [4]. The system creates and shares between the federations of nodes compact watchlists of IP addresses encoded in Bloom filters. Another P2P approach for collaborative intrusion detection is proposed by Zhou *et al.* [5]. It implements a distributed hash table (DHT) system to share detection information. Each peer submits its blacklist to a fully distributed P2P overlay. The participating nodes are notified if other peers are attacked by the same source. However, both methods use a single traffic feature, which might be too restrictive for detecting some important characteristics of large-scale intrusions.

In a distributed IDS proposed by Dash *et al.* [6], local detectors use a binary classifier to analyze incoming/outgoing host traffic and raise an alarm if a threshold value is crossed. Through their information sharing system (ISS), those alarms are sent to a random set of global detectors that generate a global view of security status of the system being monitored. DefCOM [7], which is a distributed system for DDoS mitigation, consists of three types of nodes: core, classifier, and alert generator nodes. It implements an overlay communication protocol between source, victim, and core networks to detect and block the attack at the source. One of the main drawbacks of both systems, however, is the separation of

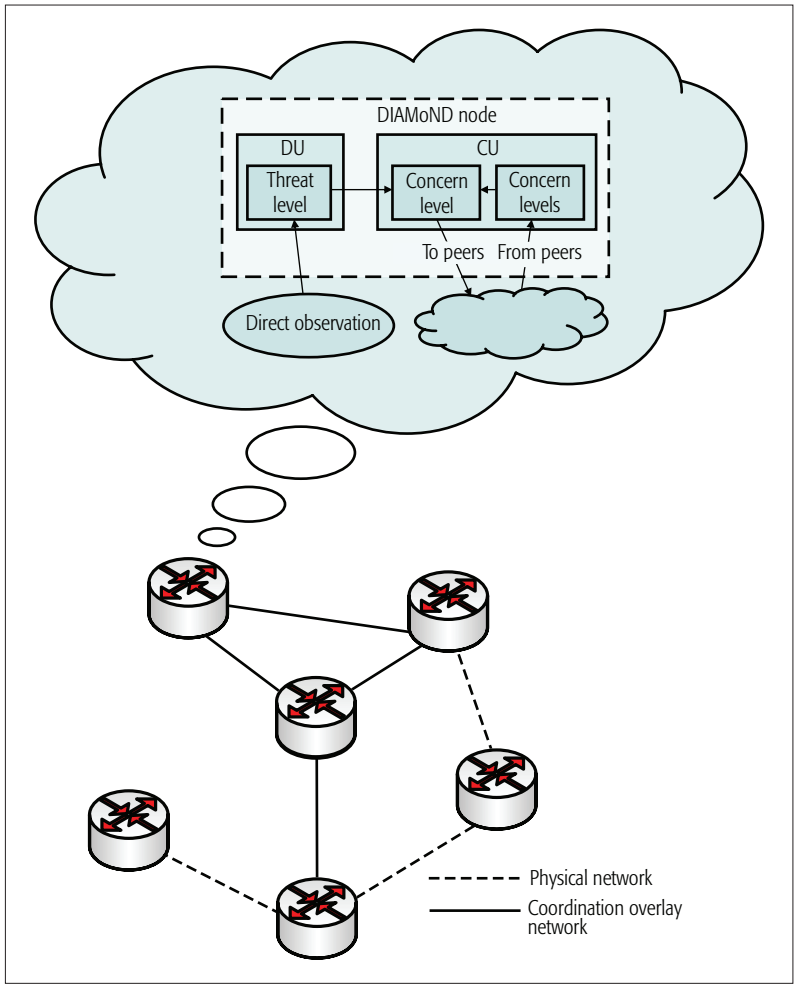

Figure 2. DIAMoND architecture.

different types of nodes and the need for the systems to coordinate messages between them.

While bio-inspired (cf. Fig. 1, e.g., [8]), and honey-bee-based algorithms in particular, are not new [9, 10], our approach is among the first to apply them to distributed-decision-driven cyber-security systems.

Honey Bee-Inspired Detection System

Forming the Analogies with Honey Bee Foraging

In honey bee foraging [2], system participants do not define the search target a priori, instead letting participants identify anomalies (resources) as they encounter them. This feature is one of the most important benefits we anticipate from adopting this bio-inspired perspective, particularly when detecting complex network attacks that might coincide with each other (in which there are no known patterns for which to look). In honey bee foragers, if enough participants identify a location as a valuable target (i.e., an anomaly), it becomes an anomaly by definition. Furthermore, as an anomaly is handled (i.e., resources are exploited), participants gradually lose interest, ceasing to identify the location as anomalous.

Another important feature of the system is that foragers who act as early scouts return to recruit additional foragers to help exploit identified anomalies (i.e., resources). They communicate not only the location, but also their "relative excitement" about the quality of the discovered resources to all other bees within range, called the foraging dance floor. This is functionally equivalent to a nonparametric description of perceived importance of the identified target, allowing very rapid and low-overhead communication and census-taking for collaborative decision making. This real-time collaborative definition of anomalies makes the system uniquely suited to discover novel targets by eliminating the need to employ any form of uniform template for comparison or recognition. We critically also adopt these features in our algorithm design.

Basing our algorithm on this system, instead of traditional distributed network anomaly detection (in which we must have a list of known patterns that indicate attacks and/or legitimate traffic), we instead allow emergent consensus to draw attention to patterns, even if some participants would not have identified the pattern as indicating an attack if assessed only independently.

System Basics

We use Distributed Intrusion/Anomaly Monitoring for Nonparametric Detection (DIAMoND): a nonparametric, fully distributed coordination framework that decouples local intrusion detection functions from network wide coordination. DIAMoND first builds coordination overlay networks on top of physical networks. DIAMoND then dynamically combines *direct observations* of traditional localized/centralized network IDS (NIDS) with knowledge exchanged with other coordinating nodes called *neighbors* to dynamically detect anomalies of underlying physical systems. Specifically, coordinating nodes in DIAMoND, analogous to honey bees, exchange generic nonparametric *levels of concern* between neighbors that reflect the observed probability of network attacks without elaborating any further details on the attacks themselves. As a result, the coordination layer of the DIAMoND framework can readily be coupled with any local detection schemes without the need for increasing the detection feature sets. The coordination network layer is also decoupled from the underlying physical network layer to facilitate flexible coordination strategies based on, for example, previously observed correlated behaviors, instead of being artificially limited to direct connectivity or geographical proximity. Interactions inside DIAMoND are limited to local neighborhood (e.g., one- or two-hop neighbors) in the overlay network, thus ensuring system scalability linear to the coordination network density instead of network size. While in general there can still be potential risks for recovery of sensitive information from the sharing of only nonparametric descriptors, in this case, since there is no need for/assumption of a uniform individual detection algorithm for local determination of level of excitement/concern across participating nodes, or even for a single node over time, no inference can be made simply from the nonparametric information shared about more sensitive features. The overall architecture of DIAMoND thus allows preservation of potentially sensitive

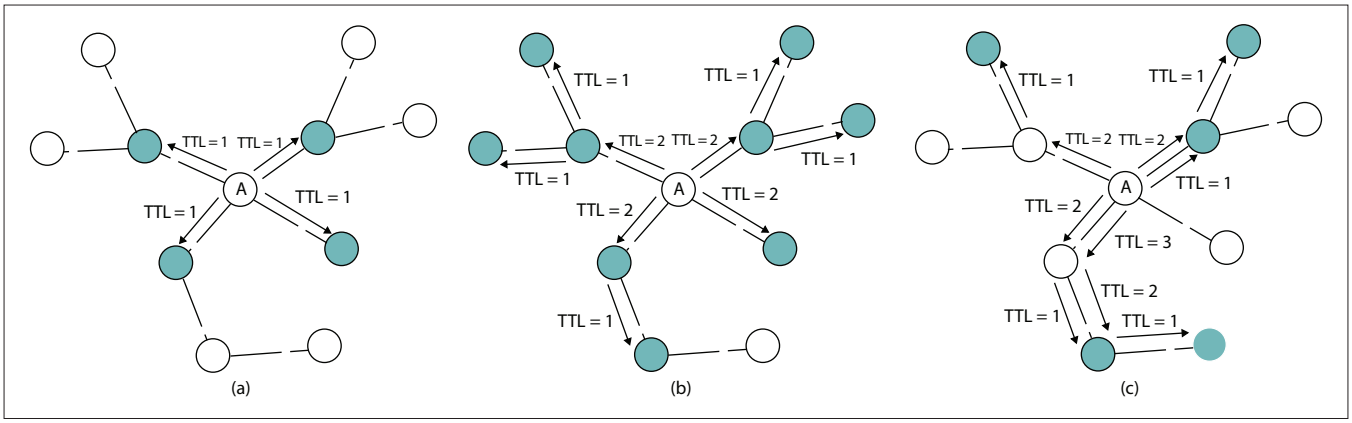

Figure 3. Neighborhood strategies: a) hop limit TTL = 1; b) hop limit TTL = 2; c) correlated attacks neighborhood.

information of individual participating parties, which eases deployment of DIAMoND across political and administrative boundaries.

System Design

Architecture Overview

DIAMoND is deployed over multiple *nodes* (switches, middleboxes) in a fully distributed architecture (Fig. 2). We define a node's *neighborhood* as a subset of all nodes with which it directly exchanges nonparametric alert-related information. Neighborhoods are dynamic and can change over time based on, for example, previously observed correlated behaviors or changes in the network topology. Two collaborating nodes enjoy a symbioti, mutual relationship, meaning both must authenticate each other and agree to join each other's neighborhoods. Furthermore, each node is equipped with two functional units: a detection unit (DU) and a coordination unit (CU). The former is responsible for the data-driven individual assessment of the so-called *threat level* — the level of likelihood that an intrusion is occurring based on the *direct observation* reported by local NIDS and/or firewall implementation. The latter calculates the *concern level*, which is a function of its own *threat level* and the *concern levels* of its neighbors (Fig. 2).

Detection Unit

Any detection or security intelligence such as NIDS or firewalls can be implemented in a DU as long as there is an appropriate plug-in to a CU to translate the output of the DU to the nonparametric threat level. Additionally, there must be an incorporated appropriate response by the DU to different levels of concerns of its neighbors (e.g., tuning of sensitivity thresholds). To foster interoperability, we do not require the extraction and provision of any potentially sensitive and/or incomparable attack details. In fact, a node may choose any local anomaly detection method independent from any other node(s), thereby making it difficult for an attacker to manipulate the local anomaly detection's influence on the CU network by making it harder to predict what types of traffic may trigger an individual, local intrusion warning. These features greatly increase the potential of such a system to be able to detect diverse characteristics of large-scale network attacks, depending on a variety of local detection algorithms adapted to DIAMoND.

Coordination Unit

Each of the participating nodes has an internal set of *sensitivity thresholds* corresponding to their "native" detection algorithms. These sensitivity thresholds are updated dynamically over time, and there is no a priori assumption of their uniformity across nodes. Since each node may employ its own local anomaly detector, these thresholds are also completely independent of each other. The sensitivity threshold is a function of the observed threat level and the level of concern of each node's neighborhood. Note that even if the sensitivity threshold is dynamic, it can be updated within a certain predefined range to prevent malicious tuning.

At each time instance, each node computes a function of the observed threat level, which is the individual data-driven assessed level of the likelihood that an anomaly is occurring.

We assign values *low*, *med*, *high* to the threat level for each node in each time instant based on the traffic observed in the local intrusion detection on that node. Values are defined such that low indicates a completely normal classification, med indicates that traffic patterns have exceeded some fixed numbers of standard deviations from normal but have not yet exceeded the rate limiting threshold to be considered an attack, and high indicates classification of a current attack by the local anomaly detector.

Each node has a level of concern at every time instant, which is a function of both the previously assessed threat level and of the total impact of the concerns of all nodes within its neighborhood computed by our naïve excitation algorithm that takes discrete values low, med, high. Values are defined such that low indicates a consensus between a node's neighbors and normal network state, med indicates that traffic patterns observed within a neighborhood have deviated from normal traffic distributions but have not yet exceeded some thresholds to be considered an attack, and high indicates classification by the node's neighborhood of a current attack.

Finally, each node determines the strength of influence of the levels of concern from its neighbors. This strength allows a node to tune preference between sensitivity and specificity provided by the collaborative network. We here

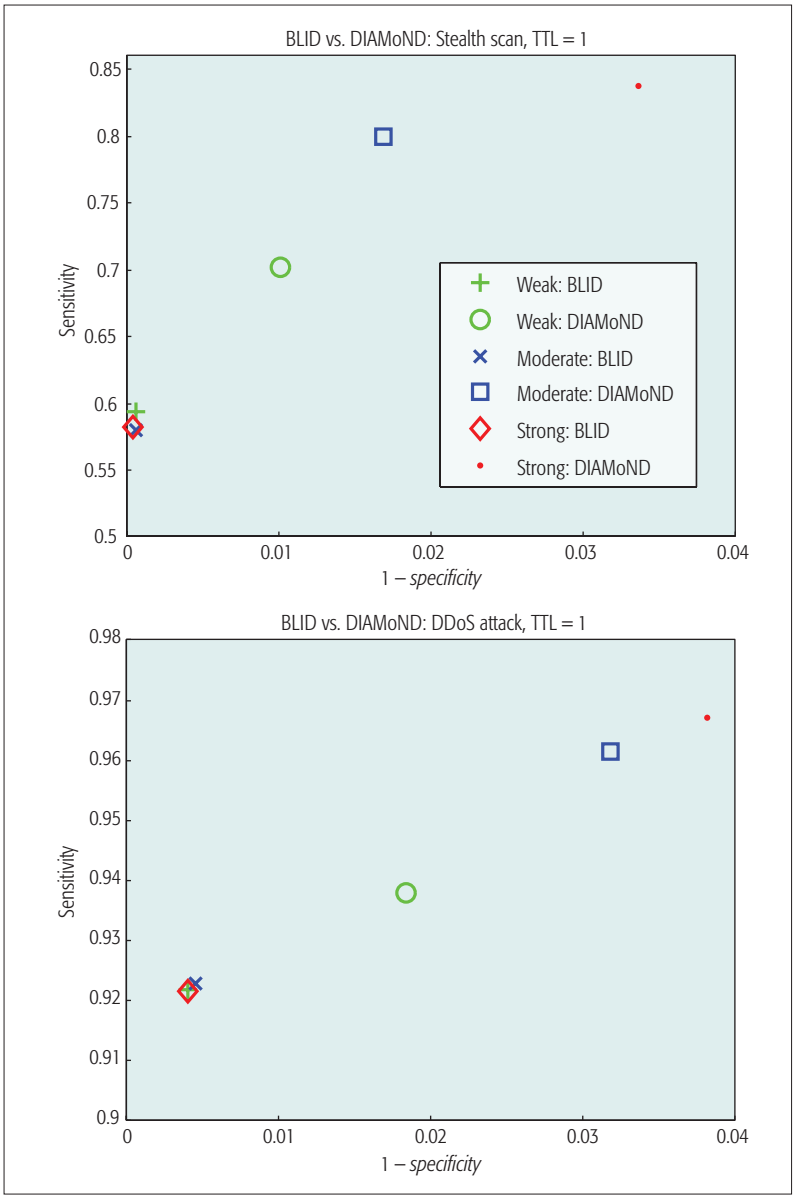

Figure 4. Comparison of DIAMoND vs. BLID for network-wide stealth scans (top) and DDoS attacks (bottom). We also explored the impact of either strengthening (strong) or weakening (weak) the influence of network neighbors to show the robustness of effect and test system sensitivity to individual-node-level detection accuracy.

present the full results for a moderate strength of influence, but results from other choices may be found in [11].

Neighborhood Strategies

Honey bees incorporate the influence of other nodes into their decision on whether or not to reinforce the signal as discussed earlier. We define and investigate two different strategies for creating the "areas" or neighborhoods to maximize the flow of meaningful information while minimizing the number of connections.

The first strategy is based on a hop limit that reflects the geographical or administrative distance between neighbors. In the simplest but very effective form, we define a neighborhood of a node by direct physical or logical connection. We also attempt to empirically verify the application of the extended neighborhoods by increasing the time to live (TTL) value (Figs. 3a and 3b). Another strategy, depicted in Fig. 3c, consists of correlating previously observed attacks and constructing neighborhoods based on the assumption that malicious activity may reoccur and be launched from the same set of compromised machines and/or against the same victims (networks, servers).

Evaluation Testbed

We have developed our prototype communication protocol as an OpenFlow controller in the POX environment[3] and evaluated it using the Mininet 2.0 network emulator.[4] Our initial software system deployment consists of 20 nodes due to computational constraints and up to 20 end-user machines connected to each node. The full specification together with the communication protocol is available to the public.[5]

In this article, we test the performance of the algorithm on an "extended star" physical topology that represents a tree of 19 links which is generated by initiating the graph with a "root" node and then attaching each subsequently created node to one of the already existing nodes in a uniform fashion.

In our experimental evaluation, we use traffic captured from the trans-Pacific line (samplepoint-F, 150 Mb/s).[6] The traffic is labeled by the MAWI working group as anomalous or normal using an advanced graph-based method that combines responses from independent anomaly detectors built on principal component analysis (PCA), the gamma distribution, the Kullback-Leibler divergence, and the Hough transform [12]. Then we develop our method based on an *X-means* algorithm. Finally, we filter all traffic labeled as anomalous by each classification method and use the remaining traffic in our benchmark traffic generator.

Each node has been equipped with a sampling detection algorithm for detecting SYN flooding attacks and TCP portscan activity [13]. The method considers TCP connections as legitimate if it samples one of multiple acknowledgment (ACK) segments (with disabled SYN flag) coming from the server. It defines two traffic features: a number of outgoing SYN segments to corresponding incoming ACK segments per source and per destination IP address. The method is combined with a rate limiting scheme — if the traffic rate is less than or equal to a predefined rate for a given IP address, it is allowed to pass the filter, whereas traffic that exceeds the rate is dropped. For the purpose of this study, we refer to the above-described algorithm as benchmark local intrusion detector (BLID). To meet the needs of our system, we extend the proposed algorithm and define the range of sensitivity rate limiting thresholds as well as the plug-in that translates the output of the algorithm to the nonparametric thread level.

We evaluate the capability of our system using two predominant attacks exploiting TCP protocol: network-wide SYN stealth scans and SYN flooding attacks that are launched from a selected percentage of the network nodes, which are considered compromised and take part in a coordinated distributed attack. For more details

[3] https://openflow.stanford.edu

[4] http://mininet.org

[5] http://mkorczynski.com/diamond.html

[6] http://mawi.wide.ad.jp/mawi

Sensitivity		1 – *specificity*		Accuracy		
BLID	DIAMoND	BLID	DIAMoND	BLID	DIAMoND	Gain
Stealth scan, TTL = 1 neighborhood						
0.58 (±0.02)	0.8 (±0.015)	$6.2e^{-4}$ ($\pm 1{,}5e^{-4}$)	0.017 (±0.003)	0.889	0.935	0.047
Stealth scan, TTL = 2 neighborhood						
0.557 (±0.021)	0.787 (±0.021)	$7.5e^{-4}$ ($\pm 5.2e^{-4}$)	0.019 (±0.003)	0.889	0.932	0.045
Stealth scan, TTL = 3 neighborhood						
0.568 (±0.029)	0.793 (±0.029)	$6.1e^{-4}$($\pm 1.7e^{-4}$)	0.02 (±0.003)	0.887	0.932	0.045
Stealth scan, attack correlation neighborhood						
0.528 (±0.027)	0.752 (±0.027)	$5.55e^{-4}$ ($\pm 1.3e^{-4}$)	0.02 (±0.003)	0.891	0.931	0.041
DDoS attack, TTL = 1 neighborhood						
0.923 (±0.012)	0.962 (±0.01)	0.005 ($\pm 7.3e^{-4}$)	0.032 (±0.004)	0.95	0.964	0.014

Table 1. Sensitivity, 1 – *specificity*, accuracy of BLID and DIAMoND, and the accuracy gain of DIAMoND over BLID. Performance at low TTL demonstrates significant benefit without increased communication overhead costs associated with higher TTLs.

Sensitivity measures the proportion of malicious packets that are correctly identified as such, specificity measures the proportion of legitimate packets that are correctly identified as such, whereas accuracy measures the proportion of correctly identified malicious and legitimate to all the packets.

on the tesbed, we refer the reader to our previous work [11].

Emulation Results

Criteria of Detection Evaluation

To assess the performance of DIAMoND, we consider three meaningful metrics: sensitivity, specificity, and overall system accuracy. Sensitivity measures the proportion of malicious packets that are correctly identified as such, and specificity measures the proportion of legitimate packets that are correctly identified as such, whereas accuracy measures the proportion of packets correctly identified malicious and legitimate to all the packets.

Also, we quantify the additional information that is gained by deploying our system on top of BLIDs. In other words, we ask by how much, if at all, the inclusion of the DIAMoND collaboration among nodes improves their accuracy relative to their use of only the local detection algorithms in isolation. In order to evaluate the information gain we use an information theoretic approach, Kullback-Leibler (K-L) divergence.

It is important to recall that the potential for improvement in accuracy is scaled by the percent of malicious packets. Since in the case of network-wide stealth scans malicious packets constitute a smaller percentage of all network traffic, the increase in accuracy is strictly bounded, meaning that, for example, 0.045 represents a substantial improvement relative to the range possible for improvement.

Detection Performance

Figure 4 shows a *sensitivity* as a function of 1 – *specificity* for network-wide stealth scans (top) and DDoS attacks (bottom) in an overlay network where neighborhoods are created on the basis of direct physical connections (TTL = 1). We present results that reflect participating nodes assigning a moderate level of influence from the concern levels of their neighbors to their own decision, but then also present results from both weakening and strengthening that influence for comparison. The results for stealth scans indicate that the more influence nodes assign to their neighbors' concern, the greater their improvement in sensitivity, without compromising specificity in comparison to BLID systems operating independently. The fact that 1 – *specificity* does not exceed 3.5 percent (in the worst case) comes from two reasons:

- Precise calibration of the rate limiting sensitivity thresholds. For example, the consensus of *level of concerns* of neighbors cannot reduce the sensitivity threshold of a chosen node below some pre-calibrated minimal value.
- The *level of concern* of a node signals the anomaly, while the decision about the assigning particular flows to *legitimate* or *malicious* classes remains with the DU.

As with the sensitivity improvements, the overall information gain of DIAMoND calculated over the accuracy of BLID increases as participating nodes increase the influence of the input from their neighbors (approximately twice as large for *moderate* and *strong* as for *weak*; Table 1).

In the evaluated attack scenarios, we observe no major distinction in the detection accuracy and information gain regardless of the neighborhood strategies (Table 1).

Finally, our results show less significant improvement in sensitivity of our system over BLID systems operating independently for DDoS attacks: between 1.6 and 4.5 percent (Fig. 4 and Table 1). We also observe that the information gain of the overlay detection system is lower (although always positive) in comparison with low-rate malicious activity, but the system can react close to the source of the attack more effectively and thereby reduce the collateral damage to a minimum.

Minimal and Marginal Deployment Gain

Deployment of networked services across administrative boundaries usually has to take place progressively. In this section, we try to understand the minimal deployment percentage

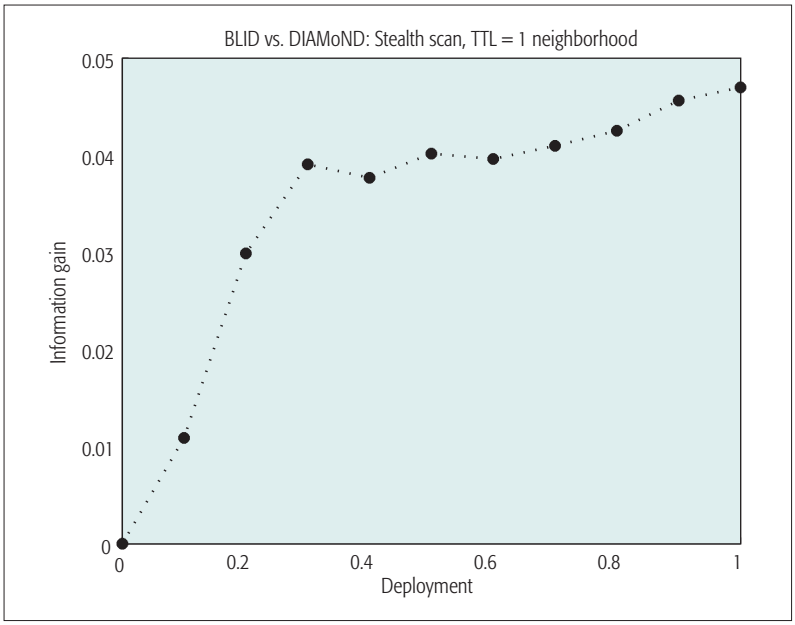

Figure 5. Information gain of DIAMoND over BLID.

needed for DIAMoND to have significant performance impact and marginal performance gain with additional deployment.

To quantitatively evaluate deployment gain, we adapt a calculation of "offline marginal utility," originally proposed to analyze the impact of additional metrics, to instead compute the incremental information gain for each additional node (relative to the information achieved with BLID). We refer the reader to some relevant literature for more details [11].

Figure 5 provides an example analysis of the deployment gain for a 20-node network under network-wide port scan probing. This figure shows a point of diminishing return such that, after 30 percent of the nodes participate in DIAMoND, the information gain is close to that achieved when all nodes are participating, and the marginal deployment gain from increasing participation is insignificant. On the other side, even when there are only 10 percent nodes participating, the information gain is already over 0.01. When 20 percent nodes are participating, the information gain reached a significant 0.03. We thus concluded that, in this case:

- Minimal effective deployment is 10 percent of the network nodes participating.
- Marginal gain is maximized at 20 percent deployment.

DIAMoND plateaus after 30 percent deployment, with minimal value gained by having additional nodes participating.

As our immediate next step we plan to explore the scalability of DIAMoND coordination protocol, and apply it to a broad set of deployment scenarios and real-network topologies.

Conclusions

In this article we investigate the potential for a self-organizing, nonparametric distributed coordination framework inspired by those observed naturally in colonies of honey bees to provide dynamic individual detection thresholds for anomalous event pattern detection on networks.

To illustrate its application, we couple DIAMoND with local anomaly detection schemes for network-wide stealthy port scan and SYN-flooding-based DDoS and evaluate its performance on an emulation testbed. DIAMoND demonstrated up to 20 percent enhancement in sensitivity without sacrificing specificity. In this article, we also systematically investigate several automated coordination neighborhood construction strategies and find that DIAMoND exhibits stable performance gain over different neighborhood strategies. This leads us to conclude that DIAMoND is robust to neighborhood size. Deployment impact shows that DIAMoND quickly reaches an information gain plateau after 30 percent of network nodes participate in coordination, which enhances the deployability of DIAMoND. It allows multiple entities, which may be functionally and/or legally prohibited from sharing cyber data, to leverage each other's insight and increase their effectiveness in cyber defense. Furthermore, DIAMoND enables real-time adaptation, eliminating the identification-designed-response delay inherent in defenses that react to known and predefined threats, and allowing active defense for emerging novel network attacks.

Acknowledgments

The U.S. Department of Homeland Security sponsored this research under the Air Force Research Laboratory (AFRL) agreement number FA8750-12-2-0232. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes. This work represents the views of the authors and does not represent official policies or endorsements, either expressed or implied, of AFRL or the U.S. Government.

References

[1] A. Noroozian *et al.*, "Developing Security Reputation Metrics for Hosting Providers," *Proc. USENIX CSET*, 2015, pp. 1–8.
[2] M. L. Winston, *The Biology of the Honey Bee*, Harvard Univ. Press, 1991.
[3] C. V. Zhou, C. Leckie, and S. Karunasekera, "A Survey of Coordinated Attacks and Collaborative Intrusion Detection," *Computers & Security*, vol. 29, no. 1, 2010, pp. 124–40.
[4] M. Locasto *et al.*, "Towards Collaborative Security and P2P Intrusion Detection," *Proc. IEEE IAW*, 2005, pp. 333–39.
[5] C. V. Zhou, S. Karunasekera, and C. Leckie, "A Peer-to-Peer Collaborative Intrusion Detection System," *Proc. IEEE ICON*, vol. 1, 2005.
[6] D. Dash *et al.*, "When Gossip Is Good: Distributed Probabilistic Inference for Detection of Slow Network Intrusions," *Proc. Nat'l. Conf. AI*, vol. 2. AAAI Press, 2006, pp. 1115–22.
[7] M. Robinson *et al.*, "DefCOM: Defensive Cooperative Overlay Mesh," *Proc. DARPA Info. Survivability Conf. and Expo.*, vol. 2, 2003, pp. 101–02.
[8] W. Mazurczyk and E. Rzeszutko, "Security – A Perpetual War: Lessons from Nature," *IT Professional*, vol. 17, no. 1, 2015, pp. 16–22.
[9] D. Karaboga and B. Akay, "A Survey: Algorithms Simulating Bee Swarm Intelligence," *Artificial Intelligence Review*, vol. 31, no. 1–4, 2009, pp. 61–85.
[10] G. A. Fink *et al.*, "Defense on the Move: Ant-Based Cyber Defense," *IEEE S&P*, vol. 12, no. 2, 2014, pp. 36–43.
[11] M. Korczyński *et al.*, "DIAMoND: Distributed Intrusion/Anomaly Monitoring for Nonparametric Detection," *Proc. IEEE ICCCN*, 2015, pp. 1–8.
[12] R. Fontugne *et al.*, "MAWILab: Combining Diverse Anomaly Detectors for Automated Anomaly Labeling and Performance Benchmarking," *Proc. ACM CoNEXT*, 2010, pp 1–12.
[13] M. Korczyński, L. Janowski, and A. Duda, "An Accurate Sampling Scheme for Detecting SYN Flooding Attacks and Portscans," *Proc. IEEE ICC*, 2011, pp. 1–5.

Biographies

Maciej Korczyński (maciej.korczynski@tudelft.nl) is a postdoctoral scientist in the cybersecurity research group at Delft University of Technology. He received his Ph.D. degree in computer science from Grenoble University of Technology, France, in 2012. Previously, he was a postdoctoral research associate at Rutgers University, New Jersey (2013–2014). His research interests include encrypted traffic classification, security of the TLS and DNS protocols,

passive and active Internet security measurements, incident data analysis, economics of cybersecurity, anomaly and attack detection, and bio-inspired cybersecurity.

ALI HAMIEH (adhamieh@gmail.com) is a research scientist at Rutgers University. He received a Ph.D. degree from the University of Versailles in France for his thesis, *Security in Wireless Ad Hoc Networks: The Cases of Jamming Attacks and Greedy Behaviors*. His research interests include network design and security.

JUN HO HUH (junho.huh@honeywell.com) is a research scientist at Honeywell ACS Labs. He received his Ph.D. in the field of cybersecurity and trustworthy computing from the University of Oxford. Since joining Honeywell, he has been involved in numerous intrusion detection projects for embedded systems, developing specification- and outlier analysis-based intrusion detection sensors for smart meters and flight controllers. His research interests also include designing intuitive cybersecurity dashboards and usable authentication solutions for control systems.

HENRIK HOLM [M] (henrik@forestglenresearch.com) is an independent consultant with Forest Glen Research, LLC. He received his Ph.D. from NTNU, Norway, in 2002. He previously worked as a postdoctoral researcher and lecturer with the University of Minnesota, and at Honeywell ACS Labs. His current interests include medical and embedded device security, signal processing, and machine learning.

S. RAJ RAJAGOPALAN (siva.rajagopalan@honeywell.com) is a research scientist at Honeywell ACS Labs. He received a Ph.D. in the field of theoretical computer science from Boston University. Since joining Honeywell in 2011, he has been leading the effort at ACS Labs of incorporating the fruits of the latest cybersecurity research into the vast portfolio of control systems in Honeywell ACS. His research interests also include using techniques from socio-cultural anthropology to address cybersecurity problems, and the study of the interactions between security and safety in modern buildings.

NINA H. FEFFERMAN (feffermn@dimacs.rutgers.edu) is an associate professor in both DEENR and DIMACS at Rutgers University. She received her Ph.D. in biology from Tufts University in 2004, and her M.S. and A.B. in mathematics from Rutgers University in 2001 and Princeton University in 1999, respectively. Her research explores evolutionary biology, epidemiology (in humans and wildlife), cybersecurity, and any other complex systems where the success of individuals involves the success of the group to which they belong.