Engineering Intrusion Detection Prevent Services for Hypervisor within Infrastructure Cloud Systems

Leonardo Vieira
University of Arkansas at Pine Bluff, Pine Bluff, Arkansas, United States, vieiral@research-cs.org

Eduardo Luque
University of Arkansas at Pine Bluff, Pine Bluff, Arkansas, United States, luquee@research-cs.org

Faculty Mentor:
Name of faculty mentor who reviewed the paper
University, City, State, Country, Email

ABSTRACT

The notion of cloud computing has evolved as an innovative computing platform, but a close examination of the paradigm, reveals it is a collection of oof the shelf components loosely connected together. Cloud computing infrastructures offers unique security challenges, with respect to ensuring that user data, and software, beyond the user's reach is secured, using highly reliable and available services. The infrastructure layer is the basis for all cloud computing enviroments and most users do not have access to it. Virtual machines infrastrucutre enable the propagation of vunerabilities, resulting in difficulties for exiting intrusion detection and prevent system (IDPS) detecting and defending against intruders. This paper outlines efforts to enhance IDPS within the hypervisor to detect multi-stage intrusion attacks within infrastructure-oriented cloud systems. This paper outlines preliminary work in contructing and testing a cloud ecosystem.

Keywords: Infrastructure as a service (IaaS) clouds, multi-stage intrusion attacks (MAS), Intrusion Detection and Prevention Systems (IDPS), Virtual Machines (VM).

1. INTRODUCTION

Cloud computing infrastructure networks have joined a large amount of essential resources like food, water, transportation, energy and many others. Cloud computing infrastructure are the natural resources which house and analyze big data in all sectors of society. These new infrastructure systems have new and unique security challenges, particulary security threats and vulnerabilities in the domain of intrusions. Therefore, the contribution of this project is the enhancement of detection, prevention mechanism for multi-stage intrusion attacks (MAS) in cloud computing environments perticulary focused on the domain of infrastructure-oriented public cloud systems, which are defined as infrastructure as a service enviroments. Currently intrusion detection and prevention system (IDPS) technologies employed in public clouds are not effective in predicting future attacks mechanisms implemented against the infrastructure, particulary by the hypervisor [3-5]. IDPSs have several limitations in cloud computing infrastructure systems such as performance, flexibility, and scalability [6-7]. The ability of an attacker to remotely exploit vulnerabilities in these systems and applications is a significant threat to virtualized machines and once an attacker compromises one element of a virtual machine other elements may also be compromised if virtualization-aware security is not implemented [8]. For example, an attacker can compromise a guest VM, which then pass the malicious code to other VMs on the same host. As result, VMs located on system increase the attack surface, risk of VM-to-VM attacks as well as, VM-to hypervisor [9-10]. Currently IDPS inference engines are inadequate to detect such malicious activity at the VM level [11-13].

2. BACKGROUND

Cloud Computing has many different models the most commonly known abstractions; Infrastructure as a Service (IaaS) is the lowest layer and it provides basic infrastructure support service, Platform as a Service (PaaS) is the environment for hosting user's application and the Software as a Service (SaaS) is the top layer which features a complete application offered as service on demand. Privacy and security is the key for success of any new technology. Cloud Computing as a new technology has some breaches compared with hard disks, hard drives, computers, usb keys and others. Many government institutions are migrating to cloud infrastructures, but concerns are rising due to the idea of sharing delicate information, which can expose private data and put a country at risk.

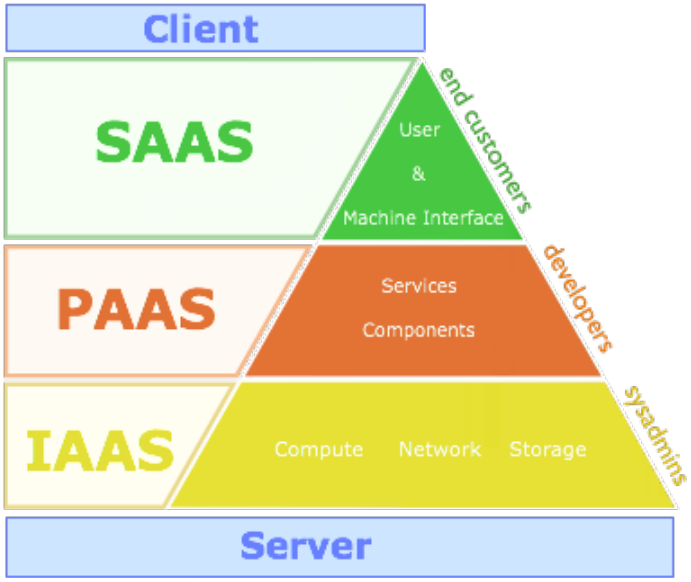

Figure 1: Cloud Architecture

3. LITERATURE REVIEW

In 2009, Kleber et al [14] proposed one of the first approaches for IDPS services in the cloud, as a middleware layer service which provided audit services which typical host based intrusion detection systems (HIDPS) and network based intrusion detection systems (NIDPS) were unable to detect. The node contains resources that are accessible through the middleware, which defines access- control policies. The service facilitated communication through the infrastructure. An event auditor monitored and captured the network data, as well as, analyzed which rules/policies were broken. The storage held behavior- based models (comparison of recent user actions to usual behavior) and knowledge-based models (known trails of previous attacks). The audited data was then sent to the IDPS service core, which analyzed the data and alarmed if an intrusion is detected. The approach differed from our approach in that it lacked one important element, reporting procedures to cloud users. In addition, the cloud clients could not customize protection procedures of the IDPS, unlike our approach.

Dastjerdi et al [15] implemented an applied agent-based approach to IDPSs within large scale computing environments. The approach basically worked by sending investigative task-specific mobile agent to every virtual host that generated similar alerts. The mobile agents could then be used to verify attacks and later assist in banning the compromised virtual machines and separating them from the network. The approach offered little customization of services, was not fully scalable to large complex infrastructures with multiple domains, similar to modern cloud infrastructures unlike our plan recognition-oriented IDPS. Bakshi et al [16] focused on protecting a cloud infrastructure from the common attack implemented against such infrastructures, that of distributed denial of service (DDoS) attacks. The approach used an installed intrusion detection system on a virtual switch and when

a DDoS attack was detected, the attacking network is blocked. The victim server is then transferred to another virtual server. In addition, the client is then blocked and other users are redirected to new virtual servers. The approach was also, not scalable to large complex infrastructures which exist today. Mazzariello et al [17] developed an approach for detecting denial of service(DoS) attacks within cloud infrastructures, against session initiation protocol (SIP). The approach was only deployed within small-scale systems and was limited to detecting o only SIP flooding attacks. The approaches were insufficient in addressing VM level attacks.

4. ADVERSARY MODEL

The project consist of a multi-cloud data center (Vulcan) which contains Eucalyptus, OpenStack and other cloud systems. VMware enabled platform virtualization allowing us to have multiple clouds on one datacenter cloud machine using a unique IP address and associated services. Our project consists of an intruder implementing multiple attacks against a single or multiple nodes within a large-scale cloud infrastructure system, where the majority of a user's data is stored within a cloud infrastructure. The user is unaware of the precise location of their data, but is only aware of the access control mechanism necessary to access the data (i.e. username/password). Each cloud system is unique with its own hypervisor, its own Xen, its own operating system and its own configuration. The project used different software like Inundator, Tor, Snort. Inundator is an anonymous intrusion detection false positives generator with ability to aim multiple targets by detecting the open ports in a network. Tor is software designed and developed by the U.S. Navy, is free software and open network that helps users defend against traffic analysis. Snort is the most famous and used open source network intrusion prevention and detection system, analyzes data flow and network traffic in real time.

The goal of this project was to implement an intrusion detection solution, augmented with plan recognition within a cloud infrastructure. This approach can provide to public cloud infrastructures a well-structured, well-organized security system capable of deployment on exiting public cloud computing infrastructures.
The intrusion detection approach was enhanced with the inclusion of intrusion prevention systems (i.e. plan recognition), which is a relatively new approach to defense networking systems, which combine the techniques of a firewall with that of a intrusion detection system, with proactive techniques[1]. These techniques prevent an attacker from entering the network by examining various data record and detection demeanor of a pattern recognition sensor, when an attack is identified, the intrusion prevention system blocks and logs the offending data [2-3].

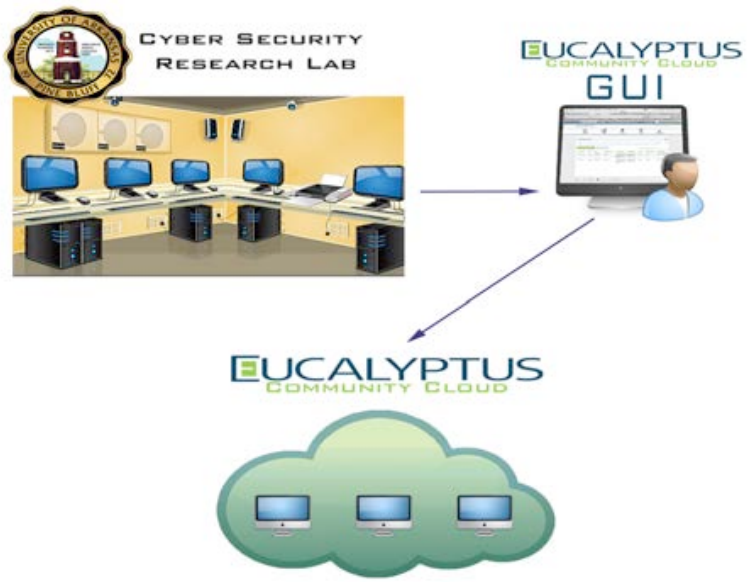

Figure 2: Eucalyptus Testbed

4.1 Cloud Infrastructure

Our cloud infrastructure architecture is designed around a three-tier hierarchy. The architecture is composed of a bottom tier, which is the node controller (NC), which is responsible for managing each virtual machine running on the physical machine. The second tier contains the cluster controller (CC). The cluster controller manages a set of node controllers residing on the same physical subnet. The final tier will be occupied by the cloud manger (CM), which manages all of the cluster controllers, and implements resource scheduling. The cloud entry point will be the cloud manager on the entire ecosystem. Our cloud system utilizes Xen virtualization technology. Xen because its hypervisor allows several guest operating systems to be executed on the same computer hardware concurrently. Thereby, allowing the partitioning of a single physical machine into multiple virtual machines to provide server consolidation and utility computing [26]. The proposed new IDPSs will be deployed within Xen hypervisor[27-41].

The ecosystem can be implemented in managed mode, with functionality similar to Amazon EC2 or Windows Azure. This allows us to implementing instances of subnetwork isolation within the system. These instances allow creating security groups, which means each user in the cloud is bound to at least one security group, similar to actual existing cloud ecosystems. In addition, users can be granted access to external networks, based on security groups controlled by the cluster controller and cloud manager. The cluster controller provide both dynamic host configuration protocol (DHCP), NA T services, the NA T provides services like elastic internet protocol (IP), which provides the means for rule configuration on the virtual network interface cards (NIC), on each virtual network/machine. Each virtual machine will be named using a scheme in which Dom0-DomN, represents the associated domains within the Xen environment. Figure 2 presents a graphical representation of the ecosystems in which a pair of virtual machines have been created, one virtual machine is within the domain of Dom0, the other two reside outside of the Dom0 domain [17].

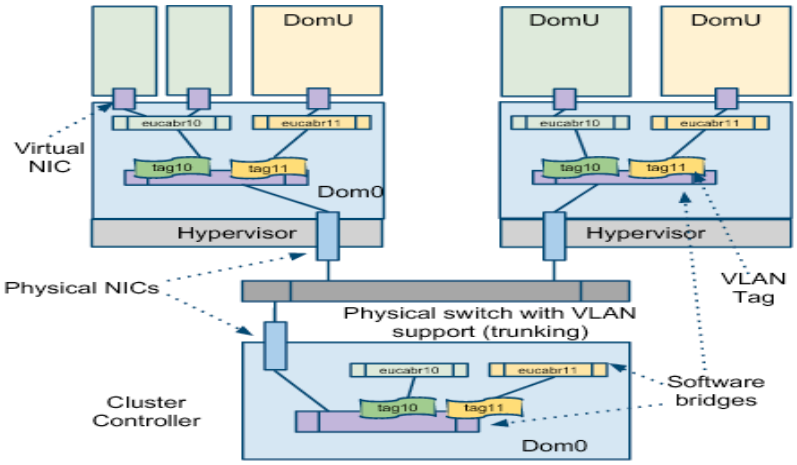

Figure 3: Cloud Ecosystem

5. MEASURING EFFECTIVENESS

There are several existing metrics that measure different aspects of IDPS, but no single metric seems sufficient to objectively measure the capability of IDPS in a cloud environment. The most commonly used metrics are based on true positive rate (TP, i.e., the probability that the IDPS outputs an alarm when there is an intrusion) and false positive rate (FP, i.e., the probability that the IDPS outputs an alarm when no intrusion occurs) [42]. Alternatively, one can use false negative (FN) rate FN = 1 - TP and true negative (TN) rate TN =1- FP. When we fine-tune an IDPS (particularly focused on MAS attacks), for example, by setting the threshold of a deviation from a normal profile there may be different TP and FP values associated with different IDPS operation points (e.g., each with a different threshold). Therefore, our approach starts with attack graphs, which are a popular representation for multi-stage attacks [43, 44]. These attack graphs are graphical representation of the different

ways multi-stage attacks can be launched against system. The nodes of these graphs depict successful intermediate attack goals with the end nodes representing the ultimate goal of an attack. The edges represent the relation that one attack goal is a stepping-stone to another goal and will thus have to be achieved before another. The nodes can be represented at different levels of abstraction, thus the attack graph representation can bypass the criticism that detailed attack methods and steps will need to be known a priori to be represented (which is almost never the case for reasonably complex systems). In our approach we will launched numerous types of denial of service attacks (DoS) and distributed denial of service (DDoS) against elements of the system. The attacks were implemented with an attack evasion tool such as inundator [45, 46], which generates an overwhelming number of FP alerts while other attacks can be performed (i.e. multi-stage attacks). The intrusion evasion minimize the possibility of detection. Also, since inundator is multithreaded, queue-driven and multiple virtual machines can be targeted at any given time. One of inundator's strengths, is its ability to read and parse snort rules, generate packets or traffic based on each rule. One of the keys to success for an IDPS is its configuration on the target machine, a good configuration will determine if our false attacks are detected or not, and multi-stage attacks. Attacks are sent anonymously via proxies using Tor [47], which is a proxy service, which will bounce our data packets around a distributed network of relays around the world, thereby, simulating actual attacks.

Table 1: Precision and Recall Parameters

	Attack = True	**Attack =False**
Detection = True	TP	FP
Detection = False	FN	TN
$Recall = \frac{TP}{TP + FN}$		$Precision = \frac{TP}{TP + FP}$

6. METHODOLOGY

Our early preliminary work examined two scenarios (1) IDPS close to the cloud controllers in the servers and (2) network IDPSs close to each of the physical machines. In the first scenario an IDPS is installed close to the cloud controllers, allowing them to sniff and analyze all traffic flowing within the cloud. Accordingly, all traffic is visible and the corresponding security groups become indistinguishable. In the second scenario an IDPS is installed close to each physical server, since each physical cloud node can potentially host numerous virtual hosts, associated with different security groups. The IDPS will have to perform security group tag removal before being able to correctly analyze each packet. The correct number, accuracy, and location of the detectors can provide an advantage to the systems owner when deploying an IDPS in cloud computing IaaS. Therefore, we concentrate on the precision and recall. Precision is the fraction of true positives determined among all attacks flagged by the IDPS. Recall is the fraction of true positives determined among all real positives in the cloud. The notions of true positive, false positive, etc. are shown in table 1. In addition we will plot the receiver operating characteristic (ROC), which is a traditional method for characterizing detector performance. It is a plot of the true positive against the false positive[48, 49].

7. CONCLUSIONS

The preliminary work described is the first step in addressing the immense unmet need for security improvements at the hypervisor-level of could computing systems. Also, the cloud ecosystem will enable us and other researchers to evaluate the trade-offs between computational overhead and granularity of analysis in terms of detection capabilities, percentage of total traffic analyzed, and cpu and memory consumption, as well as packet loss. The testbed when completed will be able to deploy multiple instances of IDPSs within the cloud ecosystem,

allowing us and other researchers to obtain multiple security observation points. In addition, the ecosystem, will provide us and other researchers with a frontend system, which operates as a NAT and traverses all traffic flows within a cloud. Therefore, the installed IDPSs will enable us to see all traffic within the virtual hosts on the cloud. Consequently, the testbed will provide numerous researchers with a quantitatively significant amount of data, on best configurations to reduce the load on each IDPS, thus reducing the possibility of packet loss. This will enable researchers to experiment with the effect of the threshold that is used in converting the conditional probability of an attack step into a binary determination. This will correspond to the practical situation that an IaaS cloud administrator has to make a binary decision based on the result of a probabilistic framework. In addition, the testbed will allow, researchers experiment with Bayesian network as perturbed by introducing variances of different magnitudes. This corresponds to the practical situation that the IaaS cloud administrator cannot accurately gauge the level of difficulty for the adversary to achieve attack goals. The impact of the imperfect knowledge can be studied through a ROC curve.

8. RESULTS AND DISCUSSION

Anomaly detection is an important aspect of any security mechanism. An ad hoc network consists of a number of peer mobile nodes that are capable of communicating with each other without a priori fixed infrastructure. However, arbitrary node movements and lack of centralized control make ad hoc networks vulnerable to a wide variety of attacks from inside as well as from outside. It is very difficult to narrow down a single node which has been attacked in a large ad hoc network. Therefore, providing effective security protection is important to ensure the continued viability of these networks in a variety of pursuits. In general, two complementary approaches exist to protect a system: prevention and detection. Intrusion prevention techniques, such as encryption and authentication, attempt to deter and block attackers. Unfortunately, prevention techniques can only reduce intrusions, not completely eliminate them. In future work, we hope use anomaly detection techniques based on Belief Networks (BNs) to identify abnormal behavior of a target feature, such as energy consumption or response time, in a computer system. Extensive simulation are planned to demonstrate that a centralized anomaly detection algorithm achieves low false alarm rates, below 5%, and high detection rates, greater than 95%. These experiments are conducted in small network environment. We will leverage this work to scale it to fit in a massive and complex computer network.

REFERENCES

[1] P. K. McKinley, F. A. Samimi, J. K. Shapiro, and C. Tang, "Service Clouds: A Distributed Infrastructure for Constructing Autonomic Communication Services," presented at the Proceedings of the 2nd IEEE International Symposium on Dependable, Autonomic and Secure Computing, 2006.

[2] P. Mell and T. Grance, "The NIST definition of cloud computing (draft)," *NIST special publication*, vol. 800, p. 145, 2011.

[3] R. Mesic, *Air Force Cyber Command (provisional) decision support*. Santa Monica, CA: RAND, 2010.

[4] M. C. Libicki, Project Air Force (U.S.). Force Modernization and Employment Program., Rand Corporation., and United States. Air Force. (2009). *Cyberdeterrence and cyberwar*. Available: http://www.rand.org/pubs/monographs/MG8 77/

[5] T. M. Wu, "Information Assurance Technology Analysis Center (IATAC)Information Assurance Tools Report – Intrusion Detection Systems," Defense Technical Information Center, McLean2009.

[6] S. T. Zargar, H. Takabi, and J. B. D. Joshi, "DCDIDP: A distributed, collaborative, and data-driven intrusion detection and prevention framework for cloud computing environments," in *Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), 2011 7th International Conference on*, 2011, pp. 332-341.

[7] A. Bakshi and B. Yogesh, "Securing cloud from ddos attacks using intrusion detection system in virtual machine," in *Communication Software and Networks, 2010. ICCSN'10. Second International Conference on*, 2010, pp. 260-264.

[8] S. Subashini and V. Kavitha, "A survey on security issues in service delivery models of cloud computing," *Journal of Network and Computer Applications,* vol. 34, pp. 1-11, 2011.

[9] S. Bhardwaj, L. Jain, and S. Jain, "Cloud computing: A study of infrastructure as a service (IAAS)," *International Journal of engineering and information Technology,* vol. 2, pp. 60-63, 2010.

[10] R. Chakraborty, S. Ramireddy, T. Raghu, and H. R. Rao, "The information assurance practices of cloud computing vendors," *IT professional,* vol. 12, pp. 29-37, 2010.

[11] A. Patel, M. Taghavi, K. Bakhtiyari, and J. Celestino JúNior, "An intrusion detection and prevention system in cloud computing: A systematic review," *Journal of Network and Computer Applications,* 2012.

[12] H. M. Alsafi, W. M. Abduallah, and A.-S. K. Pathan, "IDPS: an integrated intrusion handling model for cloud computing environment," *International Journal of Computing & Information Technology (IJCIT),* vol. 4, pp. 1-16, 2012.

[13] S. Morrow, "Data Security in the Cloud," *Cloud Computing: Principles and Paradigms, Edited by Rajkumar Buyya, James Broberg and Andrzej Goscinski Copyright,* pp. 573-592, 2011.

[14] K. Vieira, A. Schulter, and C. Westphall, "Intrusion detection for grid and cloud computing," *It Professional,* vol. 12, pp. 38-43, 2010.

[15] A. V. Dastjerdi, K. A. Bakar, and S. G. H. Tabatabaei, "Distributed intrusion detection in clouds using mobile agents," 2009, pp. 175-180.

[16] A. Bakshi and B. Yogesh, "Securing cloud from ddos attacks using intrusion detection system in virtual machine," 2010, pp. 260-264.

[17] C. Mazzariello, R. Bifulco, and R. Canonico, "Integrating a network IDS into an open source Cloud Computing environment," 2010, pp. 265-270.

[18] M. A. Ayd\, \#305, A. H. Zaim, K. G\, \#246, and k. Ceylan, "A hybrid intrusion detection system design for computer network security," *Comput. Electr. Eng.,* vol. 35, pp. 517-526, 2009.

[19] C. F. Schmidt, N. Sridharan, and J. L. Goodson, "The plan recognition problem: an intersection of psychology and artificial intelligence," *Artificial Intelligence,* vol. 11, pp. 45-83, 1978.

[20] R. Wilensky, "Planning and understanding: A computational approach to human reasoning," 1983.

[21] F. Cuppens, F. Autrel, A. Miege, and S. Benferhat, "Recognizing malicious intention in an intrusion detection process," 2002, pp. 806-817.

[22] G. Chen, H. Yao, and Z. Wang, "Research of wireless intrusion prevention systems based on plan recognition and honeypot," 2009, pp. 1-5.

[23] G. Chen, H. Yao, and Z. Wang, "An intelligent WLAN intrusion prevention system based on signature detection and plan recognition," 2010, pp. 168-172.

[24] S. Edelkamp, C. Elfers, M. Horstmann, M. S. Schr√∂der, K. Sohr, and T. Wagner, "Early Warning and Intrusion Detection based on Combined AI Methods," 2009.

[25] P. R. Cohen, C. R. Perrault, and J. F. Allen, "Beyond Question- Answering," DTIC Document1981.

[26] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, *et al*., "Xen and the art of virtualization," *ACM SIGOPS Operating Systems Review*, vol. 37, pp. 164-177, 2003.

[27] S. Axelsson, "Intrusion detection systems: A survey and taxonomy," Technical report2000.

[28] R. Di Pietro and L. V. Mancini, *Intrusion detection systems*: Springer V erlag, 2008.

[29] C. J. Fung, J. Zhang, I. Aib, and R. Boutaba, "Robust and scalable trust management for collaborative intrusion detection," 2009, pp. 33-40.

[30] P. Garcia-Teodoro, J. Diaz-Verdejo, G. Macia-Fernandez, and E. V azquez, "Anomaly-based network intrusion detection: Techniques, systems and challenges," *computers & security*, vol. 28, pp. 18-28, 2009.

[31] N. Guilbault and R. Guha, "Experiment setup for temporal distributed intrusion detection system on amazon's elastic compute cloud," 2009, pp. 300-302.

[32] K. Hwang, S. Kulkareni, and Y. Hu, "Cloud security with virtualized defense and reputation-based trust mangement," 2009, pp. 717-722.

[33] G. Isaza, A. Castillo, and N. Duque, "An intrusion detection and prevention model based on intelligent multi-agent systems, signatures and reaction rules ontologies," 2009, pp. 237-245.

[34] H. Kaur, "NETWORK INTRUSION DETECTION AND PREVENTION ATTACKS," *International Journal of Computers & Technology*, vol. 2, pp. 21-23, 2012.

[35] R. P. Lippmann, D. J. Fried, I. Graf, J. W. Haines, K. R. Kendall, D. McClung, *et al*., "Evaluating intrusion detection systems: The 1998 DARPA off-line intrusion detection evaluation," 2000, pp. 12-26 vol. 2.

[36] S. Pandey, K. K. Gupta, A. Barker, and R. Buyya, "Minimizing Cost when Using Globally Distributed Cloud Services: A Case Study in Analysis of Intrusion Detection Workflow Application," *Cloud Computing and Distributed Systems Laboratory, The University of Melbourne, Australia, Melbourne, Australia, Tech. Rep*, 2009.

[37] R. Sommer and V. Paxson, "Outside the closed world: On using machine learning for network intrusion detection," 2010, pp. 305- 316.

[38] A. Sperotto, G. Schaffrath, R. Sadre, C. Morariu, A. Pras, and B. Stiller, "An overview of IP flow-based intrusion detection," *Communications Surveys & Tutorials, IEEE*, vol. 12, pp. 343- 356, 2010.

[39] S. X. Wu and W. Banzhaf, "The use of computational intelligence in intrusion detection systems: A review," *Applied Soft Computing*, vol. 10, pp. 1-35, 2010.

[40] C. V. Zhou, C. Leckie, and S. Karunasekera, "A survey of coordinated attacks and collaborative intrusion detection," *computers & security*, vol. 29, pp. 124-140, 2010.

[41] Q. Zhu, C. J. Fung, R. Boutaba, and T. Basar, "A Distributed Sequential Algorithm for Collaborative Intrusion Detection Networks," 2010, pp. 1-6.

[42] G. Gu, P. Fogla, D. Dagon, W. Lee, B. Skori, and #263, "Measuring intrusion detection capability: an information- theoretic approach," presented at the Proceedings of the 2006 ACM Symposium on Information, computer and communications security, Taipei, Taiwan, 2006.

[43] Ver, #243, N. Mateos, #237, C. A. Villagr, #225, *et al*., "Definition of response metrics for an ontology-based Automated Intrusion Response Systems," *Comput. Electr. Eng.*, vol. 38, pp. 1102-1114, 2012.

[44] G. Modelo-Howard, S. Bagchi, and G. Lebanon, "Determining Placement of Intrusion Detectors for a Distributed Application through Bayesian Network Modeling," presented at the Proceedings of the 11th international symposium on Recent Advances in Intrusion Detection, Cambridge, MA, USA, 2008.

[45] D. J. Chaboya, R. A. Raines, R. O. Baldwin, and B. E. Mullins, "Network intrusion detection: automated and manual methods prone to attack and evasion," *Security & Privacy, IEEE,* vol. 4, pp. 36-43, 2006.

[46] M. Handley, V. Paxson, and C. Kreibich, "Network intrusion detection: Evasion, traffic normalization, and end-to-end protocol semantics," 2001, pp. 9-9.

[47] S. J. Murdoch and G. Danezis, "Low-cost traffic analysis of Tor," 2005, pp. 183-195.

[48] R. Lippmann, R. K. Cunningham, D. J. Fried, I. Graf, K. R. Kendall, S. E. Webster, *et al*., "Results of the DARPA 1998 offline intrusion detection evaluation," in *Recent advances in intrusion detection*, 1999, pp. 829-835.

[49] J. Caberera, B. Ravichandran, and R. K. Mehra, "Statistical traffic modeling for network intrusion detection," in *Modeling, Analysis and Simulation of Computer and Telecommunication Systems, 2000. Proceedings. 8th International Symposium on*, 2000, pp. 466-473.

Authorization and Disclaimer

Authors authorize LACCEI to publish the paper in the conference proceedings. Neither LACCEI nor the editors are responsible either for the content or for the implications of what is expressed in the paper.