DETECTING MALICIOUS SCRIPT IN THE REAL TIME CLOUD SERVER

NAGARJUNAPU KAMAL BABU [#1], L. SOWJANYA [#2]

[#1] MCA Student, Master of Computer Applications,

D.N.R. College, P.G.Courses & Research Center, Bhimavaram, AP, India.

[#2] Assistant Professor, Master of Computer Applications,

D.N.R. College, P.G.Courses & Research Center, Bhimavaram, AP, India.

ABSTRACT

Almost cloud services are prominent within the private, public and commercial domains. All the domains are in a critical nature regarding the storage of files or documents; therefore, security and resilience are increasingly important aspects. In this application we try to find out those files which are almost malware type ,which can cause attack to the cloud infra structure as a services. If these type of files are found the cloud server should identify them very easily and then it should be blocked by not allowing the users to download the files from that cloud server.By conducting various experiments on our proposed protocol, we finally came to a conclusion that our proposed approach is best in identifying the malware files in the cloud and block the files not to download from the cloud server for the end users.

1. INTRODUCTION

Cloud computing is the use of computing resources (hardware and software) that are delivered as a service over a network (typically the Internet). The name comes from the common use of a cloud-shaped symbol as an abstraction for the complex infrastructure it contains in system diagrams. Cloud computing entrusts remote services with a user's data, software and computation. Cloud computing consists of hardware and software resources made available on the Internet as managed third-party services. These services typically provide access to advanced software applications and high-end networks of server computers.

In this proposed application we introduce and discuss an online cloud anomaly detection approach, comprising dedicated detection components of our cloud resilience architecture. More specifically, we exhibit the applicability of novelty detection under the one-class support Vector Machine (SVM) formulation at the hypervisor level, through the utilisation of features gathered at the system and network levels of a cloud node.We demonstrate that our scheme can reach a high detection accuracy of over 90% whilst detecting various types of malware and DoS attacks.

At the infrastructure level we consider: the elements that make up a cloud datacentre, i.e. cloud nodes, which are hardware servers that run a hypervisor in order to host a number of Virtual Machines (VMs); and network infrastructure elements that provide the connectivity within the cloud and connectivity to external service users. The main motivation for designing this proposed application is all the cloud servers failed to identify and block the malware documents from the cloud server,which laid a problem for storing the sensitive data securely inside the cloud server.

The intrinsic properties of virtualised infrastructures (such as elasticity, dynamic resource allocation, service co-hosting and migration) make clouds attractive as service platforms. Though, at the same time they create a new set of security challenges. These have to be understood in order to better protect such systems and make them more secure. A number of studies have addressed aspects of cloud security from different viewpoints (e.g. the network, hypervisor, guest VM and Operating System (OS)) under various approaches derived either from traditional rule-based Intrusion Detection Systems (IDSs) or statistical anomaly detection models. This paper presents a cloud security solution derived from a sub-domain of anomaly detection, viz. novelty detection. In this section we firstly review the challenges arising from the virtualisation embedded within cloud technologies and further discuss background and related work with respect to anomaly detection in cloud environments. We also present the architectural context, within which the research presented in this paper is carried out

2. LITERATURE SURVEY

Literature survey is the most important step in software development process. Before developing the tool, it is necessary to determine the time factor, economy and company strength. Once these things are satisfied, ten next steps are to determine which operating system and language used for developing the tool. Once the programmers start building the tool, the programmers need lot of external support. This support obtained from senior programmers, from book or from websites. Before building the system the above consideration r taken into for developing the proposed system.

1) Malware Detection in Cloud Computing Infrastructures

AUTHORS: Michael R. Watson, Noor-ul-hassan Shirazi

Cloud services are prominent within the private, public and commercial domains. Many of these services are expected to be always on and have a critical nature; therefore, security and resilience are increasingly important aspects. In order to remain resilient, a cloud needs to possess the ability to react not only to known threats, but also to new challenges that target cloud infrastructures. In this paper we introduce and discuss an online cloud anomaly detection approach, comprising dedicated detection components of our cloud resilience architecture. More specifically, we exhibit the applicability of novelty detection under the one-class support Vector Machine (SVM) formulation at the hypervisor level, through the utilisation of features gathered at the system and network levels of a cloud node.

2) Data leakage detection

AUTHORS: P. Papadimitriou and H. Garcia-Molina

We study the following problem: A data distributor has given sensitive data to a set of supposedly trusted agents (third parties). Some of the data are leaked and found in an unauthorized place (e.g., on the web or somebody's laptop). The distributor must assess the likelihood that the leaked data came from one or more agents, as opposed to having been independently gathered by other means. We propose data allocation strategies (across the agents) that improve the probability of identifying leakages. These methods do not rely on alterations of the released data (e.g., watermarks). In some cases, we can also inject "realistic but fake" data records to further improve our chances of detecting leakage and identifying the guilty party.

3) Anomaly Detection: A Survey

AUTHORS: VARUN CHANDOLA, ARINDAM BANERJEE, and VIPIN KUMAR

Anomaly detection is an important problem that has been researched within diverse research areas and application domains. Many anomaly detection techniques have been specifically developed for certain application domains, while others are more generic. This survey tries to provide a structured and comprehensive overview of the research on anomaly detection. We have grouped existing techniques into different categories based on the underlying approach adopted by each technique. For each category we have identified key assumptions, which are used by the techniques to differentiate between normal and anomalous behavior. When applying a given technique to a particular domain, these assumptions can be used as guidelines to assess the effectiveness of the technique in that domain. For each category, we provide a basic anomaly detection technique, and then show how the different existing techniques in that category are variants of the basic technique. This template provides an easier and more succinct understanding of the techniques belonging to each category. Further, for each category, we identify the advantages and disadvantages of the techniques in that category. We also provide a discussion on the computational complexity of the techniques since it is an important issue in real application domains. We hope that this survey will provide a better understanding of the different directions in which research has been done on this topic, and how techniques developed in one area can be applied in domains for which they were not intended to begin with.

4) A Survey on Outlier Detection Techniques for Credit Card Fraud Detection

AUTHORS: Ms. Amruta D. Pawar

Credit card fraud detection is an important application of outlier detection. Due to drastic increase in digital frauds, there is a loss of billions dollars and therefore various techniques are evolved for fraud detection and applied to diverse business fields. The traditional fraud detection schemes use data analysis methods that require knowledge about different domains such as financial, economics, law and business practices. The current fraud detection techniques may be offline or online, and may use neural networks, clustering, genetic algorithm, decision tree etc. There are various outlier detection techniques are available such as statistical based, density based, clustering based and so on. This paper projected to find credit card fraud by using appropriate outlier detection technique, which is suitable for online applications where large scale data is involved. The method should also work efficiently for applications where memory and computation limitations are present. Here we have discussed one such unsupervised method Principal Component Analysis(PCA) to detect an outlier.

3. EXISTING SYSTEM

In the existing clouds there are many security issues that takes place which are storing and accessing the data. Also there is no enough security in all primitive cloud storage servers for storing and accessing the data to and from the remote servers. The Data Owners will try to upload the data into the cloud server in a plain text manner and the data users try to receive the data from the cloud server and during the data transfer if any un-authorized user try to inject any malicious content inside the files or try to view the files, there is no security in the current cloud servers.

LIMITATION OF EXISTING SYSTEM

1. The following are the main limitations of the existing system. They are as follows:
2. The existing cloud servers cant able to access the data in a encrypted manner rather than they are stored directly in a plain text manner.
3. All the data which is stored inside the cloud server will be directly stored into the server space despite of verifying the presence of any malware or virus content inside it.
4. There is no technique to automatically identify the presence of malware content inside the documents which is going to be uploaded or downloaded.

4. PROPOSED SYSTEM

In this proposed work we discuss the detection of anomalies using a novelty detection approach that employs the one-class Support Vector Machine (SVM) algorithm and demonstrate the effectiveness of detection under different anomaly types. More specifically, we evaluate our approach using malware and Denial of Service (DoS) attacks as emulated within a controlled experimental testbed. The malware samples used are Kelihos and multiple variants of Zeus. Here in our proposed approach we try to upload some worms or virus affected files into the cloud server through which our cloud server can easily find out the presence of such a malware files which are available in the cloud server.

ADVANTAGES OF THE PROPOSED SYSTEM

The following are the advantages of our proposed system. They are as follows:

1. The proposed cloud servers can able to access the data in a encrypted manner rather than in a plain manner.
2. All the data which is stored inside the cloud server will be directly stored into the server space and they will be verified by the cloud server if there are any malware content available in that file.
3. This proposed SVM integrated cloud computing technique can automatically identify the presence of malware content inside the documents which is going to be uploaded or downloaded.
4. The proposed technique can block the malware content files not to downloaded for the end users

5. SOFTWARE PROJECT MODULES

Implementation is the stage where the theoretical design is converted into programmatically manner. In this stage we will divide the application into a number of modules and then coded for deployment. We have implemented the proposed concept on Java programming language with JEE as the chosen language in order to show the performance this proposed protocol. The proposed application is mainly divided into 3 modules. They are as follows:

1. Data Owner
2. Data User
3. Cloud Server

5.1 Data Owner Module

This is the first module in which the data owner is one who try to upload a set of useful and sensitive documents into the un trusted cloud server. At this stage the data owner initially try to encrypt the files using RSA algorithm and then try to upload all those files into the cloud server. In this proposed application there are multiple data owners present inside the cloud in order to store their individual files into the cloud server.

5.2 Data User Module:

He is the one who gets register initially into the application and waits for the cloud server approval. Once the cloud server approve the end user ,then he/she can login into the system with all his basic credentails.Once he get login into his account he has the facility to search all the files that are available in the cloud server. So if he want to download any file from the server he need to send request to the service provider to download the data in a plain text manner. So once the file request is send to the service provider that request will be processed by the cloud server and in turn send keys for the end user for downloading the file in a plain manner. If the requested file contains no malware content ,then only the end user can download the file in a plain text manner and in turn he can download the same in his PC.If the same requested file contains any malware content ,then the file cant be downloaded by the end user even he got the decryption key from the cloud server.

5.3 Cloud Server Module:

The cloud server is one who can login with his pre-defined username and password and once he gets login ,then he can enter into his account and try to see all the files which are available in its server location. He can view list of service providers who got registered and he can also activate or de-activate the service providers. He can view the list of end users who got registered for participating into cloud data access. He can also see the list of malicious files which are available in the cloud server. He also has many other functions like ,sending alerts for the end users regarding the malicious files. He can send keys for decrypting the data and view the files by authorized users. He can block the malicious files not to be downloaded by the end users and a lot more

6. RESULTS (OUTPUT SCREENS)

Home Page

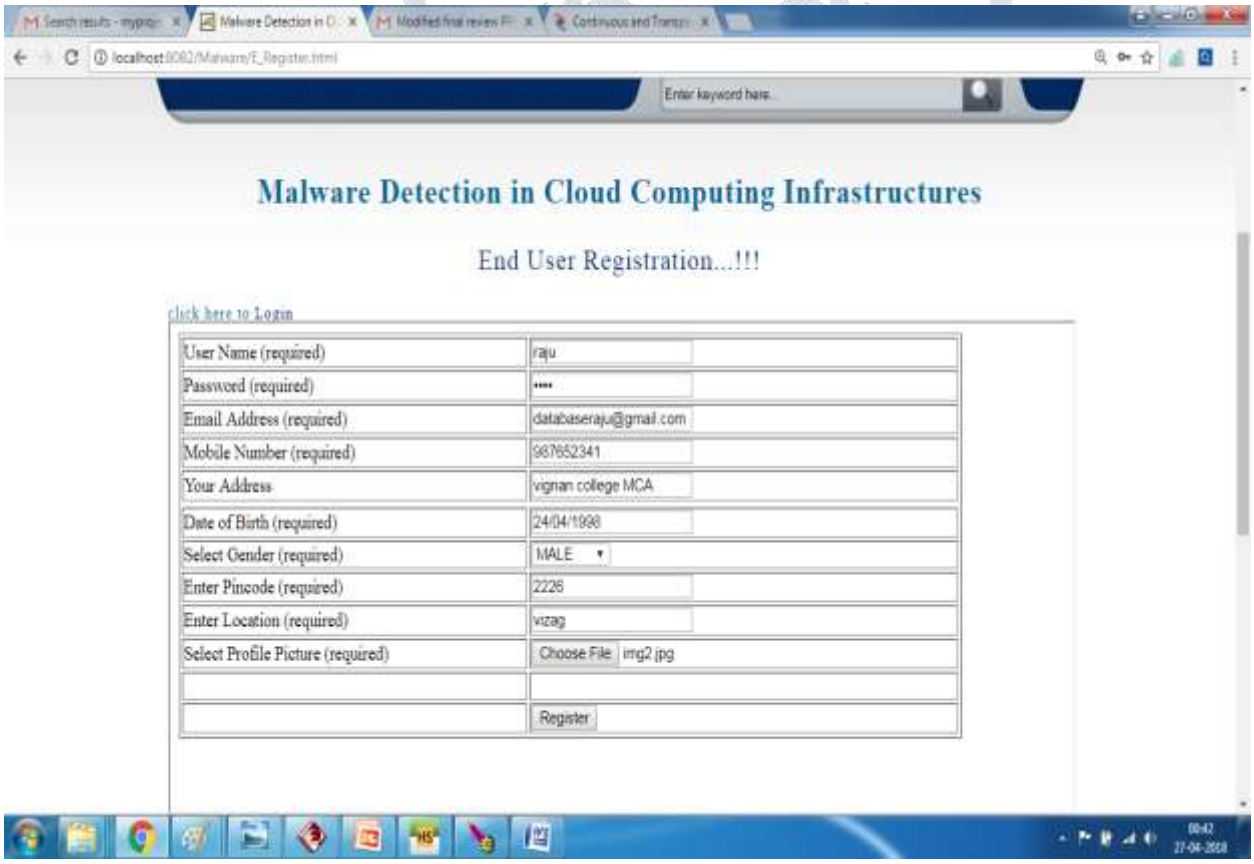
Explanation:

This is a home page which contains following links like serviceprovider,End User as well as Cloud Server and registration as well as login.

SERVICE PROVIDER REGISTRATION PAGE

Explanation: Here we can see there is a page for service provider registration.Here the service provider should fill all the fields properly if not registration will fail.

END USER REGISTRATION PAGE



Explanation: Here we can see there is a page for enduser registration.Here the end user should fill all the fields properly if not registration will fail.

CLOUD SERVER LOGIN PAGE

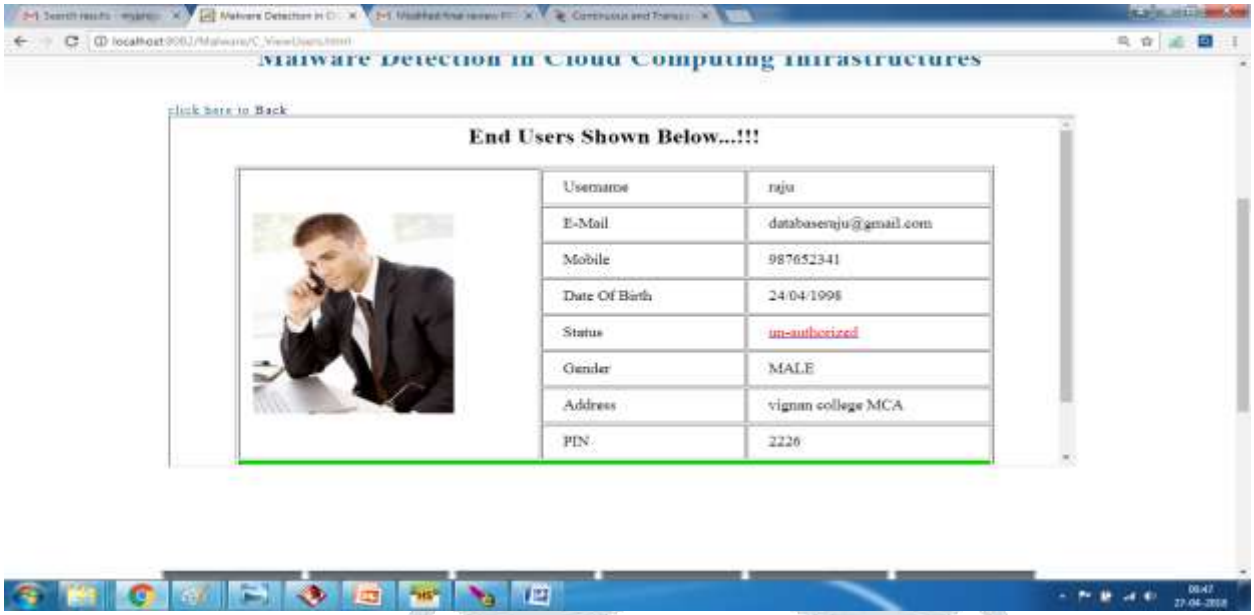
Here the cloud server login successfully if the user enter a valid id and password

CLOUD SERVER MAIN PAGE

Explanation:

He entered into his login account and he can see various facilities which are avaiable for the coud server.View Service Providers Who recently registerd and waiting for activation or authorizatiom for login

Here we can see the status as Un-authorized,which means the cloud server need to authorize the service provider for getting login into the system.View Users Who recently registerd and waiting for activation or authorizatiom for login

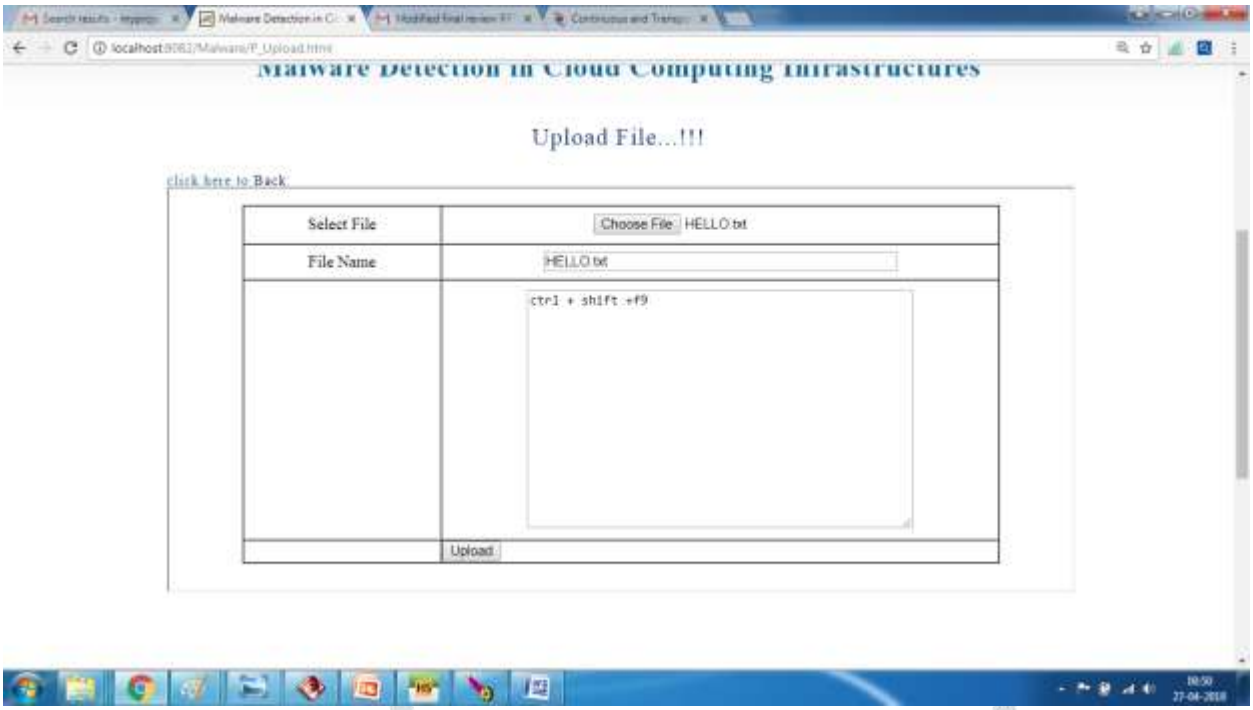


Cloud Server activates both the newly added users and Service providers, then the status will be changes as follows:

In the same way

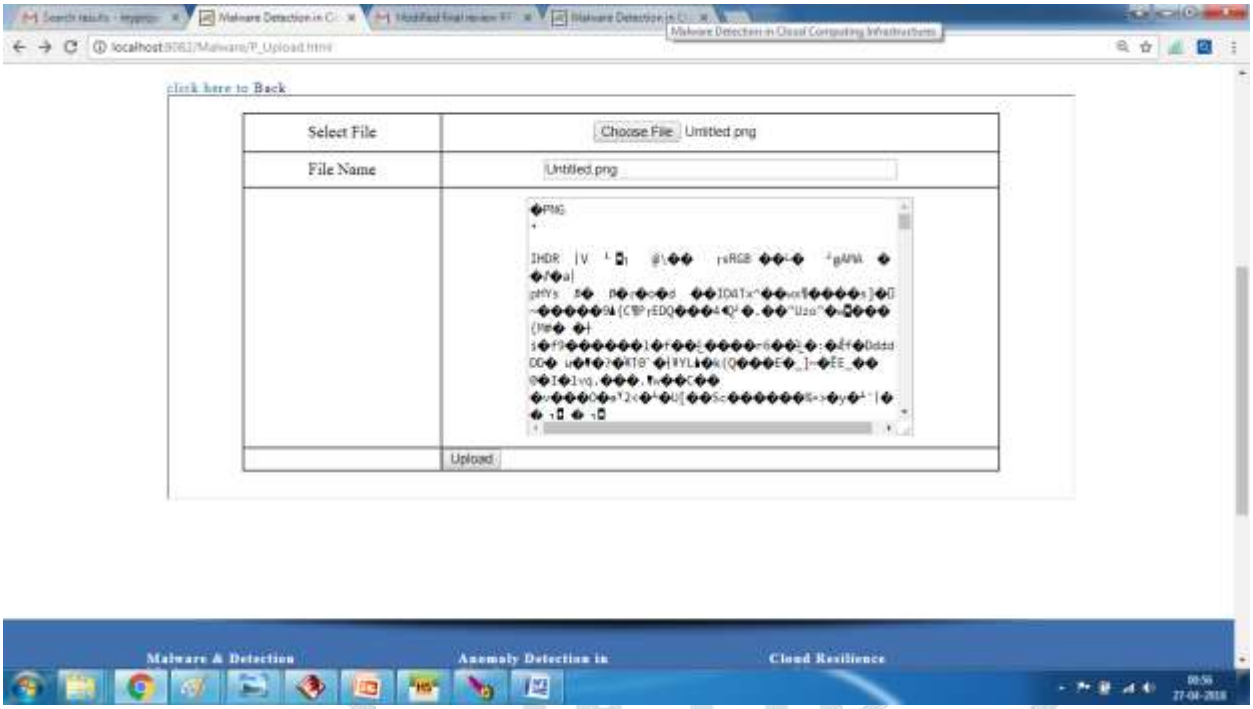
Now the service provider try to login into his account

Service provider will try to upload a file



Data Uploaded successfully as there is no blocked content or malware available

See as the file is not a proper one..The file may be under malicious one and it may be blocked by the ckoud server



Now the file is of not a valid type hence it is detected as malware

See in the admin we can see the percentage of users who try to upload the malware contents and the cloud server can block those users.Now cloud server will block the user

7. CONCLUSION

In this project we introduce an online anomaly detection method that can be applied at the hypervisor level of the cloud infrastructure. The method is embodied by a resilience architecture that was initially defined in [4], further explored in [36], [37] and which comprises the System Analysis Engine

(SAE) and Network Analysis Engine (NAE) components. These exist as submodules of the architecture's Cloud Resilience Managers (CRMs), which perform detection at the end-system, and in the network respectively. Our evaluation focused on detecting anomalies as produced by a variety of malware strains from the Kelihos and Zeus samples under the formulation of a novelty detector that employs the one-class Support Vector Machine (SVM) algorithm. Moreover, in order to empower the generic properties of our detection approach we also assess the detection of anomalies by the SAE and NAE during the onset of DoS attacks

8. REFERENCES

[1] A. Marnerides, C. James, A. Schaeffer, S. Sait, A. Mauthe, and H. Murthy, "Multi-level network resilience: Traffic analysis, anomaly detection and simulation," ICTACT Journal on Communication Technology, Special Issue on Next Generation Wireless Networks and Applications, vol. 2, pp. 345–356, June 2011.

[2] J. P. G. Sterbenz, D. Hutchison, E. K. C¸ etinkaya, A. Jabbar, J. P. Rohrer, M. Sch¨ oller, and P. Smith, "Resilience and survivability in communication networks: Strategies, principles, and survey of disciplines," Comput. Netw., vol. 54, no. 8, pp. 1245–1265, Jun. 2010. [Online]. Available: http://dx.doi.org/10.1016/j.comnet. 2010.03.005

[3] A. K. Marnerides, M. R. Watson, N. Shirazi, A. Mauthe, and D. Hutchison, "Malware analysis in cloud computing: Network and system characteristics," IEEE Globecom 2013, 2013.

[4] M. R. Watson, N. Shirazi, A. K. Marnerides, A. Mauthe, and D. Hutchison, "Towards a distributed, self-organizing approach to malware detection in cloud computing," 7th IFIP/IFISC IWSOS, 2013.

[5] M. Garnaeva, "Kelihos/Hlux Botnet Returns with New Techniques." Securelist http://www.securelist.com/en/blog/655/ Kelihos Hlux botnet returns with new techniques.

[6] H. Binsalleeh, T. Ormerod, A. Boukhtouta, P. Sinha, A. Youssef, M. Debbabi, and L. Wang, "On the analysis of the zeus botnet crimeware toolkit," in Privacy Security and Trust (PST), 2010 Eighth Annual International Conference on, Aug 2010, pp. 31–38.

[7] T. Brewster, "GameOver Zeus returns: thieving malware rises a month after police actions," Guardian Newspaper, 11, July, 2014, http://www.theguardian.com/technology/2014/jul/ 11/gameover-zeus-criminal-malware-police-hacking.

[8] A. K. Marnerides, P. Spachos, P. Chatzimisios, and A. Mauthe, "Malware detection in the cloud under ensemble empirical model decomposition," in Proceedings of the 6th IEEE International Conference on Networking and Computing, 2015.

[9] L. Kaufman, "Data security in the world of cloud computing," Security Privacy, IEEE, vol. 7, no. 4, pp. 61–64, July 2009.

[10] M. Christodorescu, R. Sailer, D. L. Schales, D. Sgandurra, and D. Zamboni, "Cloud security is not (just) virtualization security: A short paper," in Proceedings of the 2009 ACM Workshop on Cloud Computing Security, ser. CCSW '09. New York, NY, USA: ACM, 2009, pp. 97–102. [Online]. Available: http://doi.acm.org/10.1145/1655008.1655022

[11] N. Gruschka and M. Jensen, "Attack surfaces: A taxonomy for attacks on cloud services," in Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on, July 2010, pp. 276–279.

[12] Y. Chen, V. Paxson, and R. H. Katz, "Whats new about cloud computing security?" EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2010-5, Jan 2010. [Online]. Available: http://www.eecs.berkeley.edu/Pubs/TechRpts/2010/ EECS-2010-5.html

[13] G. Gu, P. Porras, V. Yegneswaran, M. Fong, and W. Lee, "Bothunter: Detecting malware infection through ids-driven dialog correlation," in Proceedings of 16th USENIX Security Symposium on USENIX Security Symposium, ser. SS'07. Berkeley, CA, USA: USENIX Association, 2007, pp. 12:1–12:16. [Online]. Available: http://dl.acm.org/citation.cfm?id=1362903. 1362915