AUTOMATIC DETECTION AND REMOVAL OF MALWARES IN CLOUD ENVIRONMENT

A.Subashini[1], R.Gowthami Saranya[2]
[1] Assistant Professor,Computer Science and Engineering, United Institute Of Technology, Coimbatore-20, Tamilnadu, India
[2] Student,Computer Science and Engineering, United Institute Of Technology, Coimbatore-20, Tamilnadu, India

ABSTRACT: Cloud services are prominent within the private, public and commercial domains. Cloud services are critical in nature; therefore security and resilience are increasingly important aspects. To become resilient, a cloud must possess to have the ability to react not only to known threats, but also to all the new challenges that approach the cloud environment. Our cloud detection and removal approach, leads to dedicated detection technique of our cloud resilience architecture. Specifically, we exhibit the applicability of malware detection with the one-class SVM formulation at the various levels, through the utilization of features gathered at the system and network levels of a cloud node. Our approach can obtain high detection accuracy on detecting various types of malwares. Our approach evaluates the data's of both the system-based data and also the network-based data depending on the various type of malwares. Finally, our approach for detection is particularly applicable to cloud environment and leads to a more flexible malware detection system capable of detecting new malware strains with no prior knowledge of their functionality or their underlying instructions.

Index Terms- Security, Malware, DES, AES, Cryptography, Dataset, Protection, anomaly, SVM

I. INTRODUCTION

Detecting malicious software is a complex problem. The ecosystem of malicious software and tools presents a terrific challenge for data administrators. Antivirus software is one of the most widely used software for detecting and stopping malicious and unwanted malicious software from the system. However, the modern malicious software means that it is increased challenging for any single developer to develop signatures for every new threat. Indeed, a recent Microsoft survey has found more than 50,000 new Malware variants of backdoors, Trojans, and bots during the year 2006 [1].

Cryptography technique translates original data into human- unreadable form. Cryptography technique is classified into symmetric key cryptography and public key cryptography (Asymmetric). This technique uses keys for translate data into human-unreadable form. So only authorized user can access data from cloud. Cipher text data's is visible for all users. Symmetric key cryptography algorithms are AES, DES, 3DES, IDEA, BRA and blowfish. The main issue is delivering the key to receiver into multi user application. These algorithm require low delay for data encode decode but provides low security. Public key cryptography algorithm is RSA and ECC algorithm. Public and private Keys are manipulated into public key cryptography algorithms. These algorithms accomplished high level security but increase delay for data encode and decode.

Steganography hide the secret data existence into envelope. In this technique existence of data is not visible to all people. Only valid receiver knows about the data existence. Text steganography technique is used to produce high security for data. Secret data of user hide into text cover file. After adding text into text cover file it looks like normal text file. If text file found by illegitimate user than also cannot get sensitive data. If illegitimate user try to recover original data than large amount of time is essential.

Overall, most of the detection techniques employed for cloud computing infrastructure are at individual layers and mostly independent of each other [2], [3].

The elements presented here form the basis in which different detection techniques can be hosted and allow the identification and attribution of malwares. In this paper we discuss the detection of malwares using a malware detection approach that employs the one-class Support Vector Machine (SVM) algorithm and demonstrate the effectiveness of detection under different malware types. More specifically, we evaluate our approach using malware and Denial of Service (DoS) attacks as emulated within a controlled experiment.

II. BACKGROUND WORKS

1. CLOUD COMPUTING

With the Todays Internet's ubiquity in modern world, many argue that some various levels of cloud computing is now a common platform. This research heavily focuses on cloud security. Cloud computing cannot be easily understood by the definition. There are many definitions, which share the same common thing: the Internet. Cloud computing is a way to access the Internet in our daily life of a single system, using all the tools and software's installed on the computers. It is also the ability to use distributed computing resources with local servers handling various applications. With the help of cloud computing users need not to worry about the location and the storage space of their data. They just start using the services anywhere and at any time. The main driver of this technology is Virtualization (Hypervisor) and virtual appliance [4].

Cloud computing offers various services that allow users to find the appropriate service that fits their infrastructure needs, Cloud service models are divided as software as a service (SaaS), Platform as a service (PaaS), and Infrastructure as a service (IaaS) [5] [6]

- Software-as-a-service (SaaS): The consumer uses the provider's applications, which are hosted in the cloud. For example, Salesforce.com CRM Application.
- Platform-as-a-service (PaaS): Consumers deploy their own applications into the cloud infrastructure. Programming languages and application development tools used must be supported by the provider. For example, Google Apps.
- Infrastructure-as-a-service (IaaS): Consumers are able to provide storage, network, processing, and other resources, and deploy and operate arbitrary software, ranging from applications to operating systems.

2. MALWARE & DETECTION METHOD

One of the biggest challenges in the cloud is the development of malware-free and secure cloud-oriented mechanisms which is related to the most adequate identification, detection and removal of malwares. This is due to the fact that, in the majority of cases, malware is the first point of initiation for large-scale Distributed Denial of Service (DDoS) attacks and email-spamming [7], [8]. Current methods of detecting attacks on cloud environment or the virtual machine (VM) resident within them do not sufficiently address cloud specific issues and problems. Despite the huge efforts employed in past studies regarding the behaviour of certain types of malware in the Internet [9], [10], so far little has been done to tackle malware presence in clouds. In particular, the studies in [11], [12] aimed to adjust the performance of traditional Intrusion Detection Systems under signature based techniques that employ Deep Packet Inspection on network packets. Moreover, work in [13], [14] studied system- related features on monitored Virtual Machine (VM) by employing Virtual Machine Introspection (VMI) methods in order to detect threats on a given Virtual Machine (VM's) Operating System.

III. MALWARE & METHODOLOGIES

1. ONE CLASS SVM

The core of our online detection methodology within the SAE and NAE lies with the implementation of the supervised one-class SVM algorithm, which is an extension of traditional two-class SVM, [23]. In practise, the one-class SVM formulation handles cases using unlabelled data (i.e., novelty detection), the main goal of which is to produce a decision function that is able to return a class vector y given an input matrix x based on the distribution of a training dataset. The class y is a binary class where one outcome is the known class, which in our case is the normal VM behaviour, and the other is the novel class, which represents any testing instances that are unknown to the classifier.

If we let $f(x) = x_1, x_2, \dots\dots., x_{n\ 1,}x_n$ represent a feature vector, then the decision function $f(x)$ takes the form:

$f(x) = \sum_{i=1}^{N} a_i\, k(x, x_i)$-p (1)

However, in order to achieve f(x) and attain the a_i multiplier over the kernel function$k(x, x_i)$ it is first required to solve the optimisation problem in Eq. (2) using Lagrange multipliers, as follows:

$\min \frac{1}{2}||\omega||^2 + \frac{1}{vn}\sum_{i=1}^{n} \varepsilon n - p$ (2)

The parameter v is extremely critical and characterises the solution by setting an upper bound on the fraction of outliers, and a lower bound on the number of support vectors. Increasing n results in a wider soft margin, meaning there is a higher probability that the training data will fall outside the normal frontier, thus identifying legitimate VM behaviour as anomalous in our case. With reference to Eq. (1), the function $k(x, x_i)$ denotes the kernel function and can be chosen to suit a particular problem. In our implementation we employed the Radial Basis Function (RBF) kernel function, which is defined as:

$k(x, x_i) = \exp(-\gamma||x - x_i||^2$ (3)

The kernel parameter g is sometimes expressed as $\frac{1}{\sigma^2}$ and a reduction in s results in a decrease in the smoothness of the frontier between normal data and outliers. It is therefore possible to produce a decision function which approximates a nearest neighbour classifier by increasing the value of γ. As we explain next, both g and n parameters are quite critical and require some tuning in order to avoid miss classifications of abnormal behaviour to normal and vice versa.

2. HYBRID COMPUTING AES-DES

In proposed algorithm (Hybrid AES-DES) the goal had been obtained by combining two algorithms called DES and AES.

A. For encryption of data

1. The input is consider as Text, image (.jpeg), audio (8 bit low level .wav file) or video (.avi) is being converted to 128 bit plain text
2. Further 128 bit text is being divided into two sets of 64 bit plain text data
3. Next this 64 bit plain text is being given as input to DES algorithm, which encrypts to give encrypted 64 bit text
4. Such two sets of encrypted 64 bit texts are then merged as single 128 bit encrypted data, which further being applied to AES algorithm for further encryption [24].

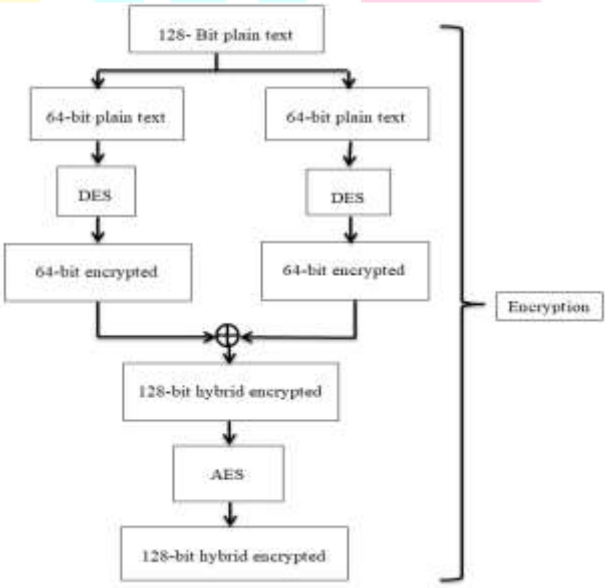


Figure 1, Hybrid AES-DES Encryption flow

B. For decryption of data:-

1. The 128 bit encrypted data is applied to AES algorithm, which provides decrypted set of 128 bit of data.
2. This one set of 128 bit of data is then further divided into two 64 bit data set.
3. These data sets are then further applied to DES algorithm to get the two decrypted set of 64 bit.
4. These two sets of 64 bit decrypted data merge into single 128 bit data.

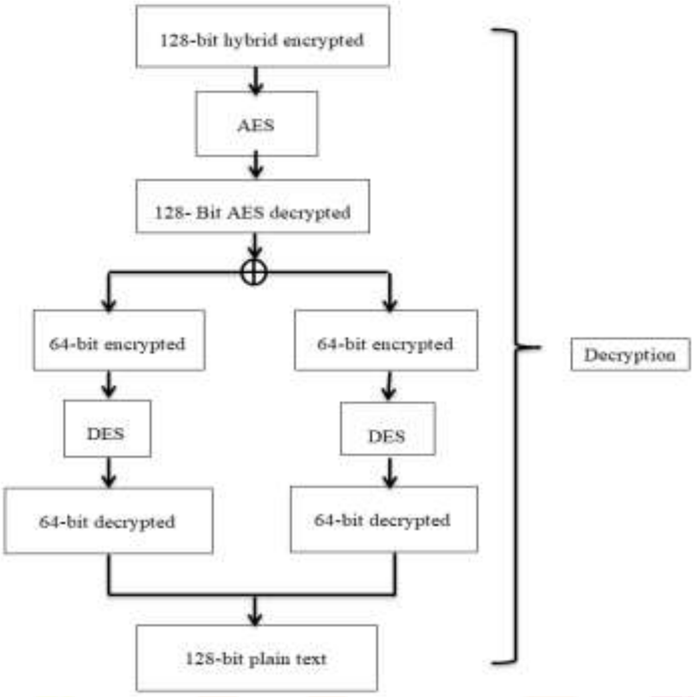


Figure 2, Hybrid AES_DES Decryption flow

IV. MODULES

A. REGISTRATION

This allows the new user to get registered in the cloud by providing the basic necessary information.

B. LOGIN

Already registered users can login to the cloud directly.

C. FILE UPLOAD WITH KEY

A file can be uploaded in the cloud with the secret key. The user cannot upload the file without the secret key. (Figure 4)

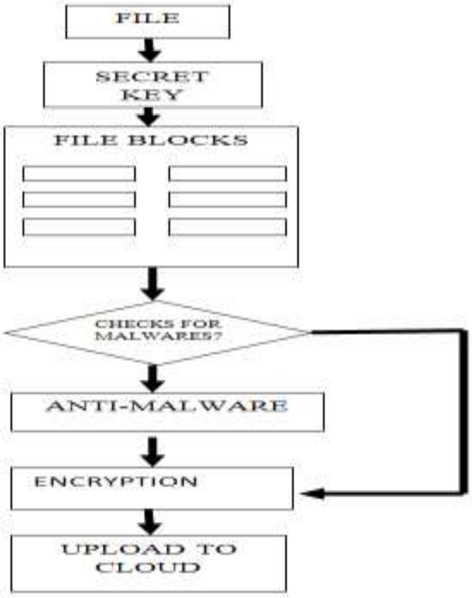


Figure 3, File uploads

D. FILE DOWNLOAD WITH KEY

The user can download their file in cloud database. The uploading key and downloading key must be same otherwise the file cannot download.(Figure_4)

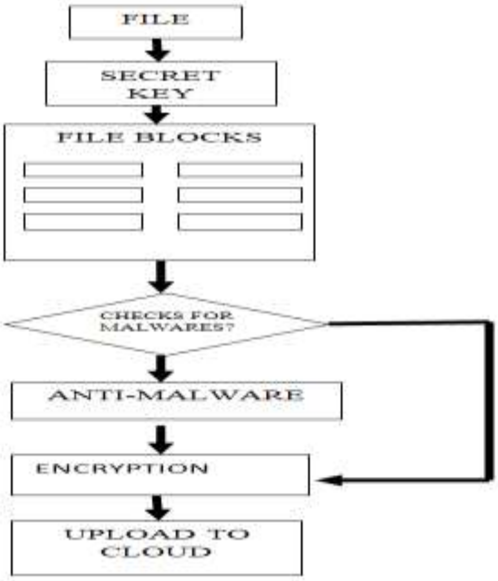


Figure 4, File Downloads

E. ENCRYPTION AND DECRYPTION

1. **Hybrid Cryptography** – Combining two cryptography techniques

It is a mode of encryption that merges two or more encryption systems. It incorporates a combination of asymmetric and symmetric encryption to benefit from the strengths of each form of encryption. These strengths are respectively defined as speed and security.
Hybrid encryption is considered a highly secure type of encryption as long as the public and private keys are fully secure.

A hybrid encryption scheme is one that blends the convenience of an asymmetric encryption scheme with the effectiveness of a symmetric encryption scheme.

Hybrid encryption is achieved through data transfer using unique session keys along with symmetrical encryption. Public key encryption is implemented for random symmetric key encryption. The recipient then uses the public key encryption method to decrypt the symmetric key. Once the symmetric key is recovered, it is then used to decrypt the message. [25]

2. **AES** (Advanced Encryption Standard)

Advanced Encryption Standard (AES) Advanced Encryption Standard (AES) algorithm not only for security but also for great speed. Both hardware and software implementation are faster still.

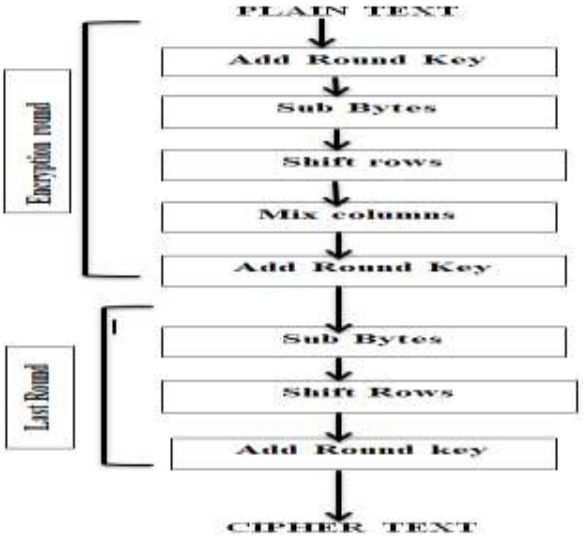
New encryption standard recommended by NIST to replace DES. Encrypts data blocks of 128 bits in 10, 12 and 14 round depending on key size. It can be implemented on various platforms especially in small devices. It is carefully tested for many security applications.
Algorithm Steps: These steps used to encrypt 128-bit block

a) The set of round keys from the cipher key.
b) Initialize state array and add the initial round key to the starting state array.
c) Perform round = 1 to 9: Execute Usual Round.
d) Execute Final Round.
e) Corresponding cipher text chunk output of Final Round Step A Study of Encryption Algorithms AES, DES and RSA for Security

1) **Usual Round**: Execute the following operations which are described above.
a) Sub Bytes
b) Shift Rows
c) Mix Columns
d) Add Round Key , using K(round)

2) **Final Round:** Execute the following operations which are described above.
Sub Bytes
a) Shift Rows
b) Add Round Key, using K(10)

3) **Encryption :** Each round consists of the following four steps: (Figure_5)
a) Sub Bytes: The first transformation, Sub Bytes, is used at the encryption site. To substitute a byte, we interpret the byte as two hexadecimal digits.
b) Shift Rows: In the encryption, the transformation is called Shift Rows
c) Mix Columns: The Mix Columns transformation operates at the column level; it transforms each column of the state to a new column.
d) Add Round Key: Add Round Key precedes one column at a time. Add Round Key adds a round key word with each state column matrix; the operation in Add Round Key is matrix addition



Figure, 5 AES Encryption

4) **Decryption :** Decryption involves reversing all the steps taken in encryption using inverse functions like (Figure_6)

   a) Inverse shift rows

   b) Inverse substitute bytes

   c) Add round key

   d) Inverse mix columns.

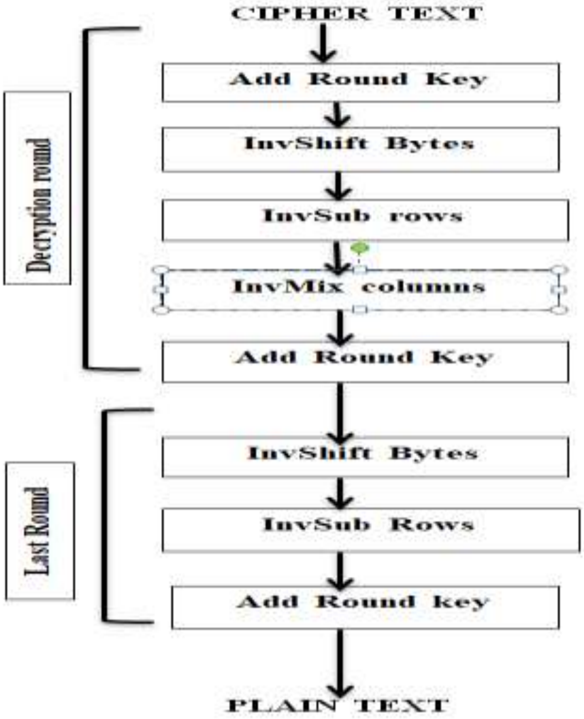


Figure 6, AES Decryption

3. **DES** (Data Encryption Standard)

Data Encryption Standard algorithm purpose is to provide a standard method for protecting sensitive commercial and unclassified data. In this same key used for encryption and decryption process [26].

DES is an implementation of a Feistel Cipher. It uses 16 round Feistel structure. The block size is 64-bit. Though, key length is 64-bit, DES has an effective key length of 56 bits, since 8 of the 64 bits of the key are not used by the encryption algorithm[27]

Algorithm Steps:

a) **Initial and Final Permutation**

The initial and final permutations are straight Permutation boxes (P-boxes) that are inverses of each other. They have no cryptography significance in DES.

b) **Round Function**

The heart of this cipher is the DES function, *f*. The DES function applies a 48-bit key to the rightmost 32 bits to produce a 32-bit output.

- **Expansion Permutation Box** − Since right input is 32-bit and round key is a 48-bit, we first need to expand right input to 48 bits.
- **XOR (Whitener).** − After the expansion permutation, DES does XOR operation on the expanded right section and the round key. The round key is used only in this operation.

- **Substitution Boxes.** − The S-boxes carry out the real mixing (confusion). DES uses 8 S-boxes, each with a 6-bit input and a 4-bit output.
- **Straight Permutation** − The 32 bit output of S-boxes is then subjected to the straight permutation

c) Key Generation

The round-key generator creates sixteen 48-bit keys out of a 56-bit cipher key.[Figure_7]

d) DES Analysis

The DES satisfies both the desired properties of block cipher. These two properties make cipher very strong.

- **Avalanche effect** − A small change in plaintext results in the very great change in the ciphertext.
- **Completeness** − Each bit of ciphertext depends on many bits of plaintext.

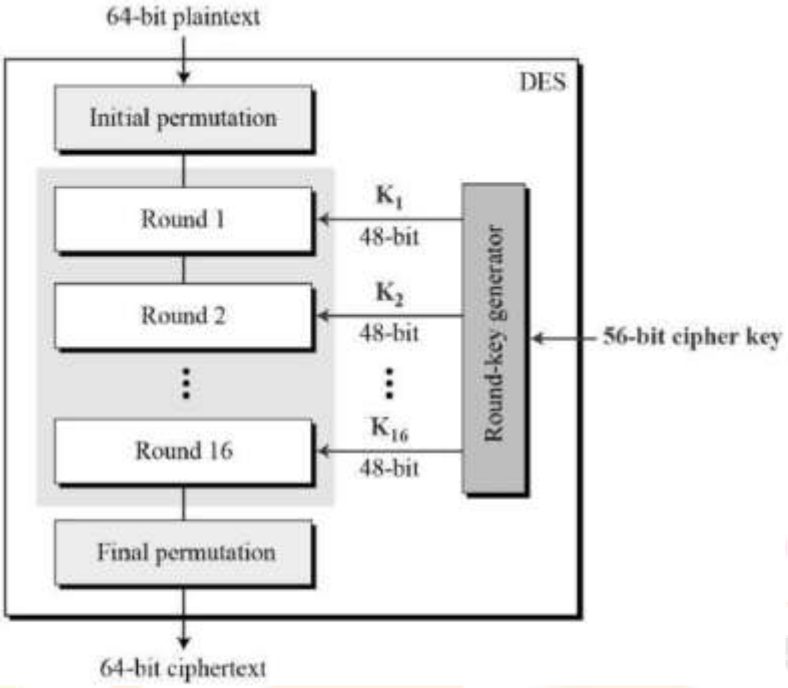


Figure 7, DES Structure

F. MALWARE DETECTION

i. Signature-based

All malware detectors basically use signature-based and virus-based techniques for detecting identities of malware in programs. There are methods using these techniques: dynamic methods that use run-time data’s of a malware, when it is executed in a memory storage; static methods, it is done by extracting features from static malware, when it is in a disk/drive, and hybrid methods that use combinations of dynamic and static techniques [21]. To identify maliciousness of a file or data using signature-based techniques, scanner tool evaluates its data to a vocabulary of virus signatures in a database or storage to see whether a signature found there are not. The advantage of using such a techniques is its effectiveness. But the main disadvantage with signature-based techniques is that they cannot protect against unknown malware [15].

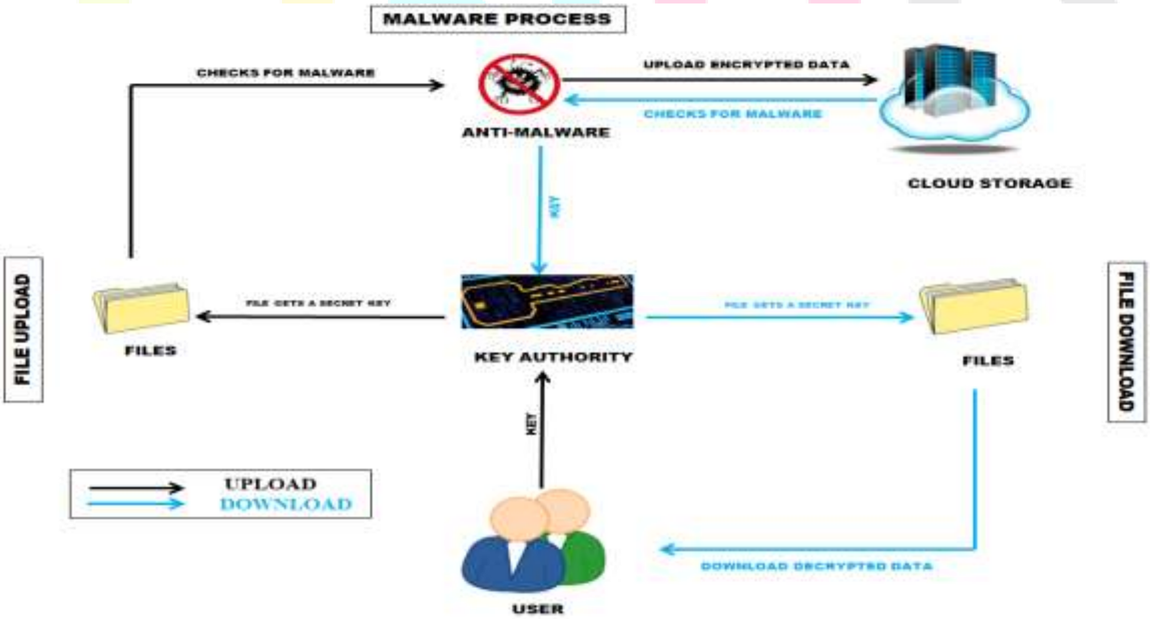


Figure 8, Architecture of the automatic detection and removal of malwares in cloud environment

ii. Anomaly-based techniques

Anomaly-based detection systems detect any kind of misusing computer or hacking the computer that fall out of the ordinary activity of a computer, while signature-based anomaly detection systems detect malwares that have a fingerprint in their databases [22, 17]. This technique detects computer's malicious software by monitoring system activities in a regular basis and classifying it as either normal or abnormal. The minor difference between signature-based and anomaly based is using classification to detect a malware, instead of using various patterns [18].

iii. Heuristic based techniques

Artificial intelligence (AI) was used with signature-based and anomaly-based techniques to enhance their efficiency. Neural networks (NNs) have been adopted for their adaptability to environmental changes and their ability of prediction [19]. Fuzzy logic is an artificial intelligence approach derived from fuzzy theory, which use approximation for logic rather than precise classical logic. Genetic algorithm is another machine learning-based technique used in malware detection process for deriving classification rules and selecting appropriate features or optimal parameters for optimum solution. It applies principles of evolutionary biology such as inheritance, mutation, selection and combination. The main advantage of this technique is the derivation of solutions from multiple directions with no need for prior knowledge about system behavior [20]. Some of the statistical and mathematical functionalities are used in malware detection by applying statistical and mathematical models on the data of system activities such as connections, bandwidth, memory usage, system etc. [16, 19]

iv. Malware detection using One Class SVM

Anomaly detection refers to the problem of finding patterns in data that do not conform to an expected behavior. Anomaly detection finds extensive use in a wide variety of applications such as fraud detection for credit cards, insurance or health care, intrusion detection for cyber-security, fault detection in safety critical systems, and military surveillance for enemy activities [28], [29], [30]. A classical approach to anomaly detection is to describe expected ("normal") behavior using one-class classification techniques, i.e. to construct a description of a "normal" state using a number of examples, e.g. by describing a geometrical place of training patterns in a feature space. If a new test pattern does not belong to the "normal" class then we consider it to be anomalous. To construct a "normal" domain we can use well-known approaches such as the Support Vector Domain Description (SVDD) [31], [32] and the One-Class Support Vector Machine (One-Class SVM) [33], possibly combined with model selection for anomaly detection [34], resampling [35], ensembling of "weak" anomaly detectors [9] and extraction of important features using manifold learning methods [36]. Both SVDD and One-Class SVM can be kernelized to describe a complex nonlinear "normal" class.

V. CONCLUSION

To conclude, we have proposed a system for combined malware detection and removal systems in cloud computing environments, all running files and malware are intercepted by submitting to one or more malware analysis tool, a complete check against a signature database to detect yet unknown malware.

We have a system of improvement in the dependence of cloud as consumers highly move towards cloud platforms for their computing needs in day-to-day life.

In this paper, we reviewed previous work on malware detection, both conventional and in the presence of memory in order to determine the best approach for detection in the cloud environment. We also argue the benefits of distributing detection throughout the cloud environment and present a new approach to coordinate detection across the cloud.

The proposal of this paper is to find the best solutions to the problems of anti-viruses and improve performance and find possible alternatives for a better working cloud environment without problems with high efficiency and flexibility.

VI. FUTURE WORKS

Future work in this field will focus on the development of detection systems based on memory, may concentrate on hardware and heuristic detection and statistical detection over the cloud environment, as opposed to signature-based detection techniques.

VII. REFERENCES

[1] Microsoft, "Microsoft security intelligence report", [online]:http://www.microsoft.com/technet/security/default.mspx, July December 2006.

[2] A. V. Dastjerdi, K. A. Bakar, and S. G. H. Tabatabaei, Distributed Intrusion Detection in Clouds Using Mobile Agents, In 3rd ADVCOMP- 09, pp. 175-180, Sliema, October 11-16, 2009.

[3] C. Mazzariello, R. Bifulco, and R. Canonico, Integrating a Network IDS into an Open Source Cloud Computing Environment, in 6th IAS-10, pp. 265-270, Atlanta, GA, August 23-25, 2010.

[4] C. Grace. "Understanding intrusion-detection systems" [J], PC Network Advisor, vol. 122, pp. 11-15, 2000.

[5] S. Subashini, V. Kavitha s.l "A survey of security issues in service delivery models of cloud computing." .Science Direct, Journal of Network and Computer Applications, pp. (1-11) January (2011).

[6] Shirlei Aparecida de Chaves, Rafael Brundo Uriarte and Carlos Becker Westphall "Toward an Architecture for Monitoring Private Clouds." S.l. IEEE December (2011).

[7] T. Brewster. (2014, Jul. 11). GameOver Zeus returns: Thieving malware rises a month after police actions, Guardian Newspaper [Online]. Available: http://www.theguardian.com/technology/ 2014/jul/11/gameover-zeus-crimina l-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 Proc. 6th IEEE Int. Conf. Netw. Comput., 2015, pp. 82–88.

[9] C. Mazzariello, R. Bifulco, and R. Canonico, "Integrating a network ids into an open source cloud computing environment," in Proc. 6th Int. Conf. Inf. Assurance Security, Aug. 2010, pp. 265–270.

[10] G. Gu, P. Porras, V. Yegneswaran, M. Fong, and W. Lee, "Bothunter: Detecting malware infection through ids-driven dialog correlation," in Proc. 16th USENIX Security Symp. USENIX Security Symp., Berkeley, CA, USA, 2007, pp. 12:1–12:16.

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

[12] M. Bailey, J. Oberheide, J. Andersen, Z. Mao, F. Jahanian, and J. Nazario, "Automated classification and analysis of internet malware," in Proc. 10th Int. Conf. Recent Adv. Intrusion Detection, 2007, vol. 4637, pp. 178–197.
[13] C. Mazzariello, R. Bifulco, and R. Canonico, "Integrating a network ids into an open source cloud computing environment," in Proc. 6th Int. Conf. Inf. Assurance Security, Aug. 2010, pp. 265–270.
[14] S. Roschke, F. Cheng, and C. Meinel, "Intrusion detection in the cloud," in Proc. 8th IEEE Int. Conf. Dependable, Autonomic Secure Comput., Dec. 2009, pp. 729–734.
[15] Ye, Y., et al., Intelligent file scoring system for malware detection from the gray list, in Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. 2009, ACM: Paris, France. p. 1385-1394.
[16] Kevadia Kaushal, P.S., Nilesh Prajapati, Metamorphic Malware Detection Using Statistical Analysis. International Journal of Soft Computing and Engineering (IJSCE), 2012. 2(3).
[17] P. García-Teodoro, J.D.-V., G. Maciá-Fernández, E. Vázquez, Anomaly-based network intrusion detection: Techniques, systems and challenges. Computers & Security, 2009 28.
[18] Bolzoni, D. and S. Etalle, APHRODITE: an Anomalybased Architecture for False Positive Reduction. 2006, Centre for Telematics and Information Technology, University of Twente: Enschede.
[19] Chandola, V., A. Banerjee, and V. Kumar, Anomaly detection: A survey. ACM Comput. Surv., 2009. 41(3): p. 1-58.
[20] Professor, S.M.B.a.R.B.V.a.A.P.a.A., Fuzzy Data Mining And Genetic Algorithms Applied To Intrusion Detection. In Proceedings of the National Information Systems Security Conference (NISSC), 2000.
[21] Idika, N. and A.P. Mathur., A Survey of Malware Detection Techniques. 2007.
[22] Teng, H.S., K. Chen, and S.C. Lu. Adaptive real-time anomaly detection using inductively generated sequential patterns. in Research in Security and Privacy, 1990. Proceedings, 1990 IEEE Computer Society Symposium on. 1990.
[23] B. Sch€olkopf, R. C. Williamson, A. J. Smola, J. Shawe-Taylor, and J. C. Platt, "Support vector method for novelty detection," in Proc. Adv. Neural Inf. Process. Syst. 12, 1999, vol. 12, pp. 582–588
[24] M.B. Vishnu & S.K. Tiong, 2008 "Security Enhancement of Digital Motion Image Transmission Using Hybrid AES-DES Algorithm", IEEE Int.
[25] Q. Y. Al-Khalidi & Prakash Kuppuswamy 2014 " Hybrid Encryption/Decryption Technique Using New Public Key and Symmetric Key Algorithm " MIS Review Vol. 19, No. 2, March (2014), pp. 1-13
[26]. Prashanti.G, Deepthi.S & Sandhya Rani.K. "A Novel Approach for Data Encryption Standard Algorithm". International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-2, Issue-5, June 2013, pp. 264.
[27]https://www.tutorialspoint.com/cryptography
[28] Manevitz, L.M., Yousef, M.: One-class svms for document classification. Journal of Machine Learning Research 2 (2001) 139–154
[29] Khan, S.S., Madden, M.G.: One-class classification: Taxonomy of study and review of techniques. The Knowledge Engineering Review 29(03) (2014) 345–374
[30] Khan, S.S., Madden, M.G.: A survey of recent trends in one class classification. In: Proceedings of the 20th Irish Conference on Artificial Intelligence and Cognitive Science, Dublin, in the LNAI. Volume 6206. Springer-Verlag (2009) 181–190
[31] Tax, D.M.J., Duin, R.P.W.: Support vector data description. Mach. Learn. 54(1) (January 2004) 45–66
[32] Chang, W.C., Lee, C.P., Lin, C.J.: A revisit to support vector data description. Technical report (2013)
[33] Scholkopf, B., Williamson, R.C., Smola, A.J., Shawe-Taylor, J., Platt, ¨ J.C.: Support vector method for novelty detection. In Solla, S.A., Leen, T.K., Muller, K., eds.: Advances in Neural Information Processing ¨ Systems 12. MIT Press (2000) 582–588
[34] Burnaev, E., Erofeev, P., Smolyakov, D.: Model selection for anomaly detection. In Verikas, A., Radeva, P., Nikolaev, D., eds.: Proc. SPIE 9875. Volume 9875. SPIE (2015)
[35] Burnaev, E., Erofeev, P., Papanov, A.: Influence of resampling on accuracy of imbalanced classification. In Verikas, A., Radeva, P., Nikolaev, D., eds.: Proc. SPIE 9875. Volume 9875. SPIE (2015) [9] Artemov, A., Burnaev, E.: Ensembles of detectors for online detection of transient changes. In Verikas, A., Radeva, P., Nikolaev, D., eds.: Proc. SPIE 9875. Volume 98751Z.
[36] Kuleshov, A., Bernstein, A.: Incremental construction of lowdimensional data representations. In: Lecture Notes in Artificial Intelligence, "Artificial Neural Networks for Pattern Recognition". Volume 9896. Springer Heidelberg (2016) 13 pp.