Cloud Based Malware Detection Technique

Sagar Shaw[1], Manish Kumar Gupta[1], Sanjoy Chakraborty[1]

[1] Computer Science & Engineering,
Institute of Engineering & Management, Kolkata
{shaw.sagar09, gupta.manish414}@gmail.com, sanjay.chakraborty@iemcal.com

Abstract. Security is one of the major concerns in cloud computing. Security is obtained to prevent threats that affect both the cloud user and cloud provider. Malicious code deployment is the main cause of threat. Many antivirus software unable to detect many modern malware threats and its enlargement in its complication has resulted in indebtedness that are being explored by malware. Apart from this Cloud computing is becoming an increasingly popular paradigm due to new services and increased media attention. This paper counsel a new model for malware detection on cloud architecture. This model enables identification of malicious and unwanted software by multiple detection engines as a result this approach provides several important benefits including better detection of malware. In this paper we use combined detection techniques, DNA Sequence Detection Process, Symbolic Detection Process and Behavioral Detection Process. The Proposed approach (PMDM) can be deployed on a VMM which remains fully transparent to guest VM and to cloud users. PMDM prevents the malicious code running in one VM (infected VM) to spread into another non-infected VM with help of hosted VMM. The main aim of this paper is to detect the malicious code by some advanced technique and warn the other guest VMs about it. A prototype of PMDM is partially implemented on one popular open source cloud architecture – Eucalyptus.

Keywords: Cloud1, Malware2, Eucalyptus 3, Antivirus4, Security5, Cloud Computing6, DNA Sequence7, Symbolic Detection and Behavioral Detection8, Sandbox9.

1 Introduction

Detecting malicious file is a complicated work. The big amount of new malware files are growing at a shocking rate. Microsoft receives over 150 thousand new unknown files each day to be analyzed. Antivirus software is one of the most widely used tools for detecting. In this paper, we suggest a new model where a file mainly undergoes these process to detect malicious behavior.

1.1 DNA Sequence Detection Process

DNA sequencing is the process of determining the precise order of nucleotides within a DNA molecule to identify regions of local or global similarity.

1.2 Symbolic Detection Process

In Symbolic detection process we cluster the files and use symbol to detect malware.

1.3 Behavioral Detection Process

One of the best way to identify malware accurately is to analyze behavior of the file. In Behavioral detection process we determine whether it is a malicious program or not using Anubis sandbox.

This Proposed Malware Detection Model is deploy into cloud architecture which gives the resultant as Cloud Deployment Model (CDM) with the help of free open source computer software Eucalyptus.

Section 2 provides background of research area of cloud technologies, security system in the cloud, malware detection etc. Section 3, we explain our Proposed System. Section 4 we show Results & Remarks of our system. Finally, section 5 Conclusions the points raised in this paper and provide some ideas for future work.

2 BACKGROUND

2.1 Cloud Computing

Cloud computing is a common term for services that offer offsite computing. There are three main types of cloud computing service models are available: [4].Software-as-a-service (SaaS), Platform-as-a-service (PaaS), Infrastructure-as-a-service (IaaS).

2.2 Security in the Cloud computing

The Security in the Cloud is provided by many companies to detect malware with industry-leading detection rates."Cipher Cloud" is company which provides service. We are providing a basic infrastructure of Cloud Malware Detection see[Fig1] [3] [9].

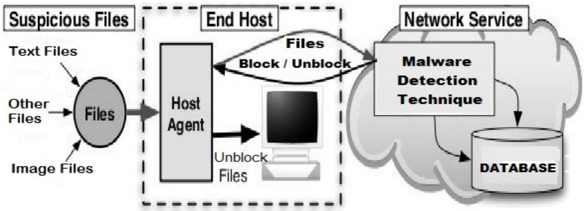


Fig 1. Cloud Malware Detection Technique.

2.3 Related Work

The traditional anti- virus software's can detect only those malware whose signatures are already present in the databases. These software's are not able to detect the new incoming malware because their signature is not present in the databases.This approach is based on the anomaly.[Fig 2] [3].

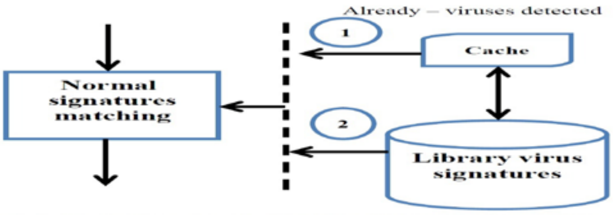


Fig 2. Traditional Anti-Virus Detection Method

3 PROPOSED SYSTEM

This paper proposes a malware detection system to be built on cloud environment, by using a series of following steps:

Initially, we will divide the system architecture into two main sections according to the mechanism of action of each part. First part, explains the Proposed Malware Detection Model (PMDM) and the second part, explains Cloud Deployment Model (CDM).

PART – I

3.1 Proposed Malware Detection Model

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

In this malware detection model, there are total three process are used to explain the mechanism,

3.1.1. Process1: DNA Sequence Detection Process.

3.1.2. Process2: Symbolic Detection Process consist of Clustering and Symbolic detection.

3.1.3. Process 3: Behavioral Detection Process using sandbox testing.

All of them are explained below in detail see [Fig 3].

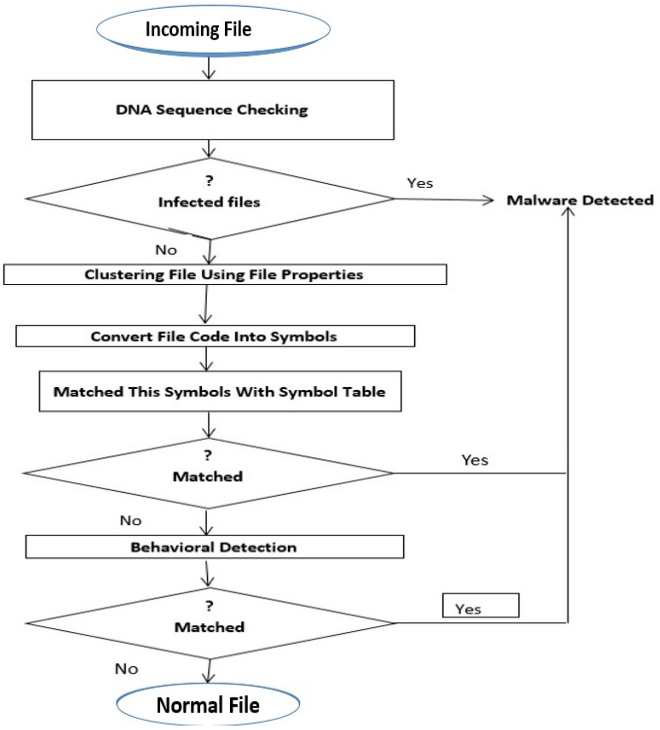


Fig 3. Flow Chart for Detection Method

In Process 1 we go for DNA Sequence checking, the initial step is the extraction of DNA sequence from a file is done by converting the file into its binary form and change each two corresponding bits into a DNA sequence character by using [Table II] [2]. The conversion is completely reversible.

TABLE II. DNA SEQUENCE MAPPING TABLE

Binary Bits	DNA Character
00	T
01	G
10	C
11	A

After that Malware_Sequence_Database is created using these steps shown in Fig 4.

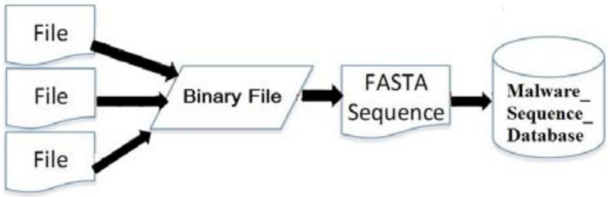


Fig 4. Creating BLAST Database.

The various input files are converted into binary files which are converted into FASTA sequence and are merged into a single FASTA sequence file called Malware_Sequence_Database. These Sequence are shown in Fig 5.

```
>c: /user/name.txt
GTAGGGCCCGTTTGGCCAAAAATTTTTTTT.
```

Fig 5. DNA Sequence

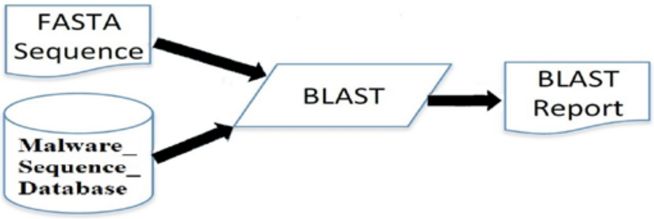


Fig 6. Comparing FASTA Sequence with Malware_Sequence_Database.

Once the Malware_Sequence_Database has been created, we are comparing for the similarities among the FASTA sequence file and Malware_Sequence_Database using Blast online software. The result of this comparison is a BLAST report, determine that the file is malicious or not see [Fig 6].

In Process 2 the files which are pass through the first process are only go for the second process. In this process first we cluster the files according to their file format, by only checking the file format see [Fig 7]

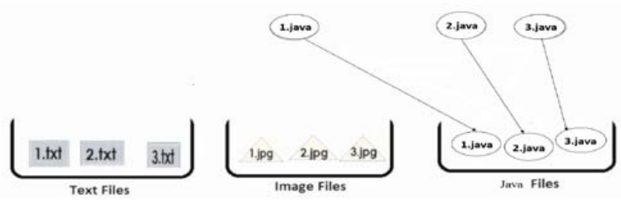


Fig 7. Clustering File Formats

After that we use symbolic detection technique in which the files are convert to symbol called symbol file. Then after we match symbol file with the existing symbol database table which contain symbol of conventional malware signatures, see [Table III].

TABLE III. SYMBOL DATABASE TABLE

SL No.	String	Symbol
1.	:A	◄
2.	Start	↕
3.	goto	§
4.	explorer	‼
5.	shutdown	¿
6.	taskkill	×
7.	md	≡
8.	>nul	Ñ
9.	off	Ö
10.	-m	…
11.	%random%	‰
12.	explorer.exe	⌂
13.	>>%random%.bat	☼
14.	For(;;)	^^^

If the file symbols are not matched with the existing database table then the file may be malicious or may not be. Otherwise the file symbols are matched with the existing database table and we can say that the file is malicious and blocked for the third process.

In Process 3 the files which are pass through the second process are only go for the third process. In this process we detect malicious files using a virtual machine that extensively used for this type of analysis by testing and running the file into a sandbox gives an optimal result to detect malware. For this purpose we use Anubis sandbox [7] which is free available. Anubis interact with file using API call and check the behavior of the file to identify whether it contain malware or not.

PART - II

3.2 Cloud Deployment Model

The Proposed Malware Detection Model (PMDM) discuss in Part-I is deploy into cloud architecture i.e. Cloud Deployment Model (CDM) by using a free open source computer software Eucalyptus. The purpose of this CDM is to implement PMDM in real cloud environment. In this experiment we determine the fitness of the proposal into the Eucalyptus architecture. PMDM is valuate against known and unknown malicious attacks. Here the PMDM system is partially implemented in the Eucalyptus architecture due to the architectural (infrastructure) limitations. The PMDM is mainly used to detect the malicious code based on the above discussed processes. And keep them as a set of warnings in a dedicated thread storage pool and block the malicious file to enter into a Guest VM or Guest Operation System see [Fig 8] [4] [5] [6].

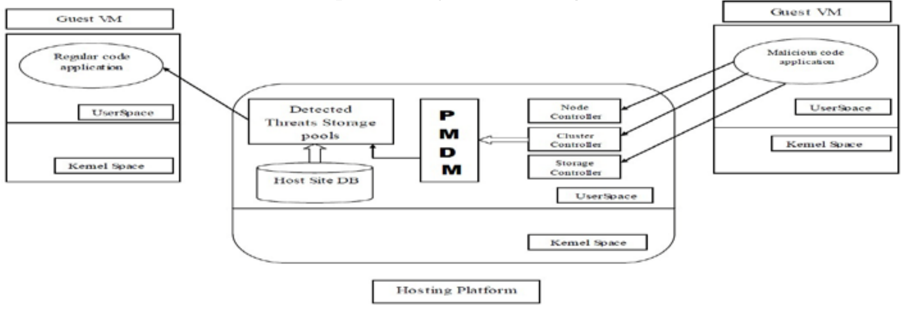


Fig 8. Proposed Malware Detection Method (PMDM) Embedded In Cloud Deployment Model (CDM)

4 RESULT ANALYSIS

This section shows how this proposal copes with attacks of malicious files in a cloud environment.

4.1 *Result of DNA Sequuence Process.*

Several experiments were designed to evaluate the usefulness of this process including document gathering, modification of DNA sequences, database creation, and software identification.

4.1.1 *Document Gathering*

A set of 1020 files of five different types were collected to test the DNA Sequence process. The files included text, image, binary, java and executable files. The types and counts of file were recorded in [Table IV].

TABLE IV. FILE COUNTS

186 Text files	220 Executable Files
169 Java Files	99 Binary Files
152 Image Files	194 HTML Files

4.1.2. *Modify DNA Sequence*

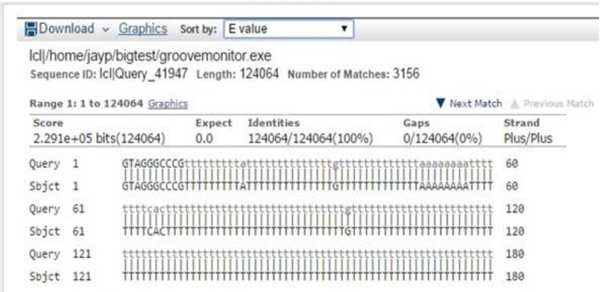
Converting the file into its binary form and change each two corresponding bits into a DNA sequence character see [Table II].

4.1.3. *Database and software*

Merged FASTA sequence file create database see [Fig 4].

The result which comes out from Process 1 is obtain by using BLAST software. We analyze the above files and observe the result one of the result output is given below:

Groovemonitor.exe malware on BLAST and found a 100% identities matched i.e. it is a malware, see results in [Fig 9(a)].



```
lcl|/home/jayp/bigtest/groovemonitor.exe
Sequence ID: lcl|Query_41947  Length: 124064  Number of Matches: 3156

Range 1: 1 to 124064 Graphics
Score                  Expect  Identities               Gaps            Strand
2.291e+05 bits(124064)  0.0    124064/124064(100%)      0/124064(0%)    Plus/Plus

Query  1     GTAGGGCCCGtttttttttatttttttttttttttgtttttttttttttaaaaaaaatttt  60
             ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  1     GTAGGGCCCGTTTTTTTTTATTTTTTTTTTTTTTTGTTTTTTTTTTTTTAAAAAAAATTTT  60

Query  61    ttttcactttttttttttttttttttttttttttttgttttttttttttttttttttttt  120
             ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  61    TTTTCACTTTTTTTTTTTTTTTTTTTTTTTTTTTTTGTTTTTTTTTTTTTTTTTTTTTTT  120

Query  121   tttttttttttttttttttttttttttttttttttttttttttttttttttttttttttt  180
             ||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||
Sbjct  121   TTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTTT  180
```

Fig 9 (a). Descriptive Analysis Result of groovemonitor.exe

Same Analysis process is applied on an image file which is malware free and it result is below 70% as expected .From the above various files results the groovemonitor.exe file is blocked and the Image File1.bin and other files are pass for the second process.

4.2 Result of Symbolic Detection Process

For symbolic detection there is also several operations were designed to evaluate the usefulness of this process including file clustering, converting characters into symbols and matching symbol with symbol table database.

4.2.1. *File Clustering.*

In file clustering a set of different types were taken and group them into a single group according to their file format see [Fig 7].

4.2.2 *Converting File Characters into Symbols*

The conversion of file characters into symbol is done by using [Table III].

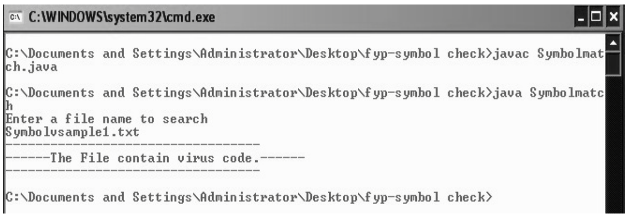
4.2.3 *Matching Symbol with Symbol Table Database*.

The matching of symbols with symbol table is done by using the following pseudocode.

```
Take symbolfile name as Input
Scan symbolfile
While not EOF do
        If symbolfile match with virus symbol Then
                Print "File contain virus"
        Else
                Print "File does contain virus"
        Endif
End While
```

Fig 10. Pseudocode for Matching Symbol with Symbol Table Database.

Files match are found in the symbol table, as example here symbolvsample.txt file and it is blocked after that and other files that not matched are pass for the third process.



```
C:\Documents and Settings\Administrator\Desktop\fyp-symbol check>javac Symbolmat
ch.java

C:\Documents and Settings\Administrator\Desktop\fyp-symbol check>java Symbolmatc
h
Enter a file name to search
Symbolvsample1.txt
-----------------------------------
------The File contain virus code.------
-----------------------------------

C:\Documents and Settings\Administrator\Desktop\fyp-symbol check>
```

Fig 11. Symbolic Detection Result of symbolvsample.txt

4.3 *Result of Behavioral Detection Process*

The files which pass the second process are only entertain in this process like from the above result all the files are entertained excluding groovemonitor.exe and symbolvsample.txt files. All the files are the input to the Anubis sandbox after that Anubis will check the files using API call and also check the behavior of the file to identify whether it contain malware or not.

4.4 *Comparison Between Traditional Malware Detection Vs Proposed Malware Detection.*

Table V. COMPARISON BETWEEN TRADITIONAL MALWARE DETECTION VS PROPOSED MALWARE DETECTION.

Features	Traditional Malware Detection Method	Proposed Malware Detection Method
Effected by security attacks	More	Less
Process time	More	Less
Implementation	Expensive	Cheap
For large number of files	Decrease Performance	Increase Performance
Attackers Intrusion	Easy	Difficult
Cloud solution	May or may not be available	Available

4.5 Advantages

4.5.1 Advantage of DNA Sequencing

The benefit of using DNA Sequencing detection method is that we can detect malware without opening the file i.e. by matching only the DNA sequence of a file form a database, we can say that a file is malicious or not. It will save time because we not need to see the whole file content to detect a malware.

4.5.2 Advantage of Smbolic Detection Process.

In symbolic detection first we cluster the file which give a benefit of Post infection protection i.e. we actually know which portion of file content we have to see exactly for malware detection. After that matching the converted symbol of file with symbol table database, will increase time efficiency as we not required to see for the whole malware signature matching only a small part of traditional malware signature symbol is sufficient to detect malware [10] [11].

4.5.3 Advantage of Behavioral Detection Process

In Behavioral malware detection solve the problem of cannot cope with malware variants i.e. malware can change their code and compiler setting to bypass the detection or Zero day protection problem i.e. once a new malware is produced its signature or symbol is unknow, so by testing the malicious file into a sandbox we can say that it is malicious or not. The overall advantage of the system is that it increases the efficiency and effectiveness for detection of malwares [10] [11].

5 CONCLUSION AND FUTURE WORK

To conclude, it has proposed an effective and advanced cloud security method that can detect the different malware attacks during cloud communication. It is also partially implemented on a popular architecture of Eucalyptus with modified version i.e. Cloud Deployment Model (CDM). This paper discusses the basic techniques in brief needed for the development of Proposed Malware Detection Model (PMDM).This technique is actually cheap, requires less processing time and provides good performance for large numbers of files compare to other traditional malware detection systems. It is totally transparent to the user. We used the both optimal traditional detection methods and modern era methods to detect malwares, like signature based detection (traditional) and DNA Sequencing method (modern era).The proposal of this work is to find the best solutions to the problems of anti-malwares and improve performance and find possible alternatives for a better working environment without problems with high efficiency and flexibility.

In future, we will see an increase in the dependence of cloud computing. Cloud technologies have become possible because of shearing physical server resources between multiple virtual machines (VMs). The advantages of this approach include an increase in the number of clients that can be served for every physical server.

References

[1] G. E. Dahl, J. W. Stokes et al., "Large-scale malware classification using random projections and neural networks", 2013 IEEE International Conference, pp. 3422 - 3426 , 31 May 2013.

[2] Jay Pedersen, Dhundy Bastola, et al., "BLAST Your Way through Malware Malware Analysis Assisted by Bioinformatics Tools", International Conference on Security and Management 2012, 2011.

[3] Safaa Salam Hatem, Dr. Maged H. wafy, et al., "Malware Detection in Cloud Computing", International Journal of Advanced Computer Science and Applications (IJACSA), vol 5, Science and Information, 2014.

[4] Dan C. Marinescu, "Cloud Computing: Theory and Practice", MK Publication, 2013.

[5] Johnson D, Kiran Murari, et al., "Eucalyptus Beginner's Guide- UEC Edition", v1.0, 25 May 2010.

[6]Hiren Parmar, L. D. C. E, GTU, et al., "Comparative Study of Open Nebula, Eucalyptus, Open Stack and Cloud Stack".

[7]Thomas Mandl, Ulrich Bayer, et al., "ANUBIS ANalyzing Unknown BInarieS The automatic Way", Virus Bulletin Conference 2009.

[8] Jon Oberheide, Evan Cooke, et al., "CloudAV: N-Version Antivirus in the Network Cloud", 17th conference on Security symposium, pp- 91-106.

[9] Mark Graham, “Behaviour of Botnets and Other Malware in Virtual Environments”, The Open Web Application Security Project 2014.

[10] https://www.youtube.com/watch?v=fV5kED7nryw.