International Innovative Research Journal of Engineering and Technology

ISSN NO: 2456-1983

International Conference on Emerging Innovation in Engineering and Technology

ICEIET-2017

Intrusion Detection System in Cloud Computing Using Mapreduce

V. Vijayalakshmi, V. Vinoth, V. Bhuvaneshwari
Senior Assistant Professor, PG Student
Christ College of Engineering and Technology, Puducherry-605010
vivenan09@gmail.com, vvinoth777be@gmail.com, Bhuvi.esswar90@gmail.com

ABSTRACT

Cloud computing involves delivering hosted services over the internet. These services are broadly divided into three categories: infrastructure-as-a-service, platform-as-a-service and software-as-a-service. Cloud computing security issues are important issue for users to access the cloud resources. Intrusion detection system is a type of security management system for computers and networks. Mapreduce has become increasingly popular as a powerful parallel data processing model. To deploy mapreduce as a data processing service over open system such as cloud computing we must provide necessary security mechanisms. This system uses an approach called hirschberg algorithm, divide and conquer approach for measuring the identity between two sequences in order to reduce time and space complexity. The proposed system is able to stop the sql injection attacks and provide rights to get access the cloud resources

KEYWORD: cloud computing, mapreduce, sql injection, hirschberg algorithm.

1. INTRODUCTION

Mostly, the researches of intrusion detection area are involved in the false positive and false negative of Intrusion Detection System (IDS). Cloud computing is the collection of virtualized and scalable resources, capable of hosting application and providing required services to the users with the "pay only for use" strategy where the users pay only for the number of service units they consume [2]. By using parallel and distributed computing, there are several benefits for resolving above problems, such as analyzing huge data sets, high performance for reading access and fault tolerance. In this paper, we propose IDS in cloud computing using mapreduce (ICCM) to stop SQL injection and provide rights to access cloud resources.

The rest of the paper is organized as follows: Section 2 reviews related works and Section 3 distinguished the new approach from previous solutions and introduced the overall system architecture. Section 4 contains experimental. Section 5 explains the result and Section 6 concludes the paper.

2. RELATED WORKS

2.1 *Intrusion Detection System [2]*

An IDS analyzes information about the activities produced from networks and seeks for malicious behavior. An extensible IDS architecture for distributed cloud infrastructure is available [13]. Detection methods are used by intrusion detection systems in two different ways, according to two different criterions: anomaly detection [4], [10], [11], [12] and misuse detection. In anomaly detection systems, a "normal profile" should be built by historical data about a system's activity and then use this profile to identify patterns of activity. On the contrary, misuse detection systems are based on specific attack signatures that are matched against the stream of audit data seeking for that malicious attack is occurring. Most of IDS are based on this misuse detection system, such as Snort. Snort is the most popular IDS especially in open source network intrusion detection systems. It can absolutely promote intrusion prevention system. Snort utilizes a rule-driven language, containing the benefits of signatures and anomaly. Snort has become the standard for the industry and many experiments on academic paper are based on it.

2.2 *Cloud Computing [2]*

The term "Cloud Computing" means the usage of computer technology (Computing) based on Internet (Cloud). The computing capabilities are provided as a service without knowledge or expertise support. Cloud Computing is the next natural step in the evolution of on demand information technology services and products. Cloud Computing became famous in October 2007 when IBM and Google announced collaboration [14]. This was

followed by IBM's announcement of the "Blue Cloud" effort. Until now, Google is one of the leaders in this technology and has built Internet consumer services like search, social networking, Web e-mail and online commerce that use Cloud Computing. The companies such as Yahoo and Amazon [5] also provide great Cloud Computing applications, too.

2.3 *Mapreduce [6]*

MapReduce is a software framework produced by Google to support parallel computations over large (multi-Petabytes) datasets on clusters of commodity nodes. Google MapReduce combines two classes of functions: *map* and *reduce*. [6] These functions are defined with respect to data structured in (key, value) pairs. *Map* takes a pair of data with a type and returns a list of key-value pairs:

map(k1, v1) -> list(k2, v2).

The *map* function is applied in parallel to every item in the input dataset. This produces a list of *(k2, v2)* pairs for each call. The framework collects all pairs with the same key from all lists and groups them together, creating one group for each key *k2*. The *reduce* function is then applied in parallel to each group, which in turn produces a collection of values:

reduce(k2, Union_List(k2,v2)) ->List (v3).

The returns of all *reduce* functions are collected as the desired result set. A simple example of MapReduce is a program that counts the appearances of each unique word in a set of unstructured documents. In this program, the *map* function will emit <*word, countInDocument*> pairs for each word that occurs in each document. The *reduce* function for each word will receive partial counts from each document, sum up the partial counts, and emit the final result.

```
map(String key, String value):
// key: document name
// value: document contents for
each word w in value:
EmitIntermediate(w, "1");

reduce(String key, Iterator values):
// key: a word
values: a list of

for each v in values: result
+=                ParseInt(v);
Emit(AsString(result));
```

The map function emits each word plus an associated count of occurrences. The reduce function sums together all counts emitted for a particular word.

Parallelism: map() functions run in parallel, creating different intermediate values from different input data sets. reduce() functions also run in parallel, each working on a different output key, all values are processed *independently*. Bottleneck: reduce phase can't start until map phase is completely finished.

Fault Tolerance: Master detects worker failures. Re-executes completed & in-progress map() tasks. Re-executes in-progress reduce() tasks. Master notices particular input key/values cause crashes in map(), and skips those values on re-execution.

There are many implementations of mapreduce available. The first java open source implementation of mapreduce is Hadoop.

2.4 *Hirschberg Algorithm [3]*

Hirschberg's algorithm is a divide and conquer version of the Needleman-Wunsch algorithm. Hirschberg's algorithm is commonly used in computational biology to find maximal global alignments of DNA and protein sequences and this proposal incorporated this algorithm here to find out the similarities.

Hirschberg's algorithm is a generally applicable algorithm for finding an optimal sequence alignment. One application of the algorithm is finding sequence alignments of DNA or protein sequences. This algorithm mainly used for reducing the time and space complexity of the system.
For example, x and y are strings to be compared, where |x| = n and |y| = m, Hirschberg's algorithm is a clever modification of the Needleman-Wunsch Algorithm which takes O(nm) time, but needs only O(min{m,n}) space for comparing these two x and y strings. The formula to plot the values is shown in the following table.

F(i,j)	A	C	C	T	G
A	F(i-1,j-1)	F(i,j-1)+p) p – gap penalty			
T	F(i-1,j)+p p – gap penalty				

Figure 1. Hirschberg Algorithm

F(i,j)= Max{F[i-1,j-1)+t(xi,yi), F(i,j-1)+px, F(i-,j)+py)}

F(i,j) 1 if(xi=yi)

0 if(xi≠yi)

t(xi,yi) - score for aligning the characters at positions i and j p is the penalty for a gap.
F(i,j) is a type of running best score as the algorithm moves through every position in the matrix. But in our approach, gap penalty has ignored. If Xi=Yi, then plot "1" else "0" till all the character has been visited.

3. PROPOSED SYSTEM

Cloud computing security issues are important issue for users to access the cloud resources. Intrusion detection system is a type of security management system for computers and networks. Mapreduce has become increasingly popular as a powerful parallel data processing model. To deploy mapreduce as a data processing service over open system such as cloud computing we must provide necessary security mechanisms.

This system uses an approach called Hirschberg algorithm, divide and conquer approach for measuring the identity between two sequences in order to reduce time and space complexity. The proposed system is able to stop the sql injection attacks and provide rights to get access the cloud resources.

There are three modules involve in this proposed system. The modules are as follows

1. Client authentication
2. IDS
3. Cloud System

3.1 *Client authentication*

Client should be authenticated one to keep the services secure. To access the cloud services the clients should be authenticated. In cloud environment at the same time many client or node can access the services available in the cloud. To access these more system request the proposed system used mapreduce concept to access the large amount of data in parallel.

3.2 *IDS*

IDS contain two parts. These are Data Mapper and the Data Reducer. Master get request from different nodes and send to home agent to requested node. Home agent provides system information and request information to master. Master sends this information to mapper. Mapper uses Hirschberg Algorithm to compare these request parameter values with signature based module. If it matches set the value as 1 for particular key otherwise 0.

Hirschberg's algorithm is a divide and conquer version of the Needleman-Wunsch algorithm. Hirschberg's algorithm is Hirschberg's algorithm is a generally applicable algorithm for finding an optimal sequence alignment. One application of the algorithm is finding sequence alignments of DNA or protein sequences. This algorithm mainly used for reducing the time and space complexity of the system.

For example, x and y are strings to be compared, where |x| = n and |y| = m, Hirschberg's algorithm is a clever modification of the Needleman- Wunsch Algorithm which takes O(nm) time, but needs only O(min{m , n}) space for comparing these two x and y strings. The formula to plot the values is shown in the following table. t(xi,yi) - score for aligning the characters at positions i and j p is the penalty for a gap. F(i,j) is a type of running best score as the algorithm moves through every position in the matrix. But in our approach, gap penalty has ignored. If Xi=Yi, then plot "1" else "0" till all the character has been visited. The predefined parameters are stored in signature based module. Mappers combine the request queries with the parameters stored in signature based module.

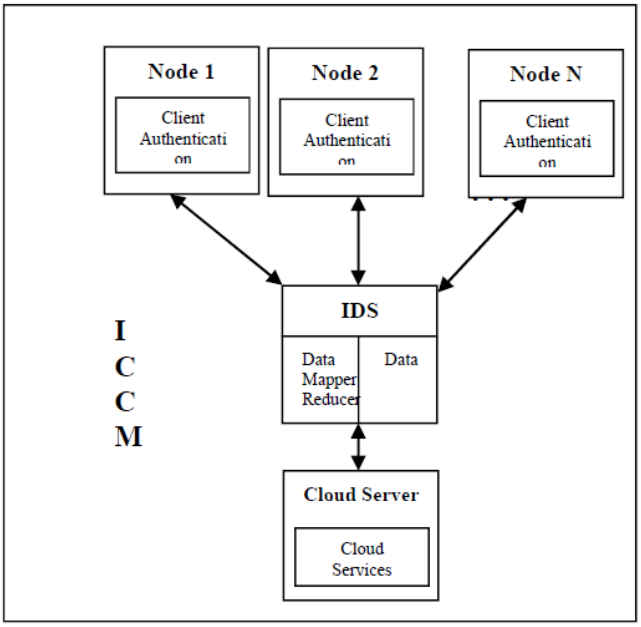

Figure 2. Architecture diagram of ICCM.

After mapper finishes its work master invoke the reducer to do reduce function. Reducer get all the output of mapper then aggregate the values for particular request. If all the values are 1 then that particular request of node is authorized node otherwise it will generate alarm as authorized node. Analyzer informs to master the information of authorized node. Authorized node can access any services which are present in cloud.

3.3 *Cloud Server*

ISSN NO: 2456-1983

Cloud system module is used to store all the cloud services in cloud environment. There are many services available in the cloud environment. These services are dynamic services.

- Email services
- Salary computation
- Excel generation
- File storage
- Xml generation
- System cleaner

Figure 3. Screenshot of Cloud Server.

These services are dynamic the client can access the services if they registered for particular services otherwise they cannot access.

4. EXPERIMENTS

Thus a perfect system for the detection of intrusion in cloud computing environment in JAVA language and MYSQL has been developed. Our proposed system effectively detects unauthorized nodes, and the results have been compared with the existing approaches. The time and space complexity are also comparatively reduced and the speed is increased. Since the proposed system is implemented in the open-source, platform-independent, object-oriented language Java it is also portable. Moreover the system is user-friendly and can also be used by naïve users without any help.

5. RESULTS

Our proposed system effectively detects unauthorized nodes, and the results have been compared with the existing approaches. The time and space complexity are also comparatively reduced and the speed is increased.

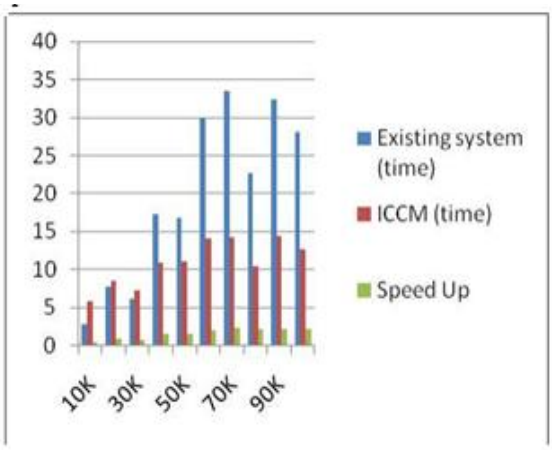

Figure 4. Experimental result for time complexity and speed.

	Existing system	ICCM
Space need to run	O(m,n)	O(min(m,n))

Figure 5. Experimental result for space complexity. Complexity.

1. Time complexity – The ICCM uses the mapreduce concept that is mapreduce is used to process the large amounts of data in parallel.
2. Speed – Since the proposed system ICCM has low time complexity the speed also increased due to process data in parallel.
3. Space complexity – The ICCM uses the Hirschberg algorithm which is used to reduce the space complexity comparing to other techniques. Space need to run the existing system is O(m,n) and space need to run the ICCM is O(min(m,n)). That means ICCM need only the minimum space to run the application.

6.CONCLUSION AND FUTURE WORK

Cloud computing security issues are important issues for users to access the cloud resources. The proposed system get numerous requests from different users at the same time to intrusion detection system, home agent collect the system information and request information to

master, then the master divides into sub problems. Mappers get these sub problems and use Hirschberg's algorithm to compare the input parameters with predefined signatures to check the particular request is authorized one or not. Hirschberg's algorithm is a generally applicable algorithm for finding an optimal sequence alignment. Reducers aggregate the output values. Analyzer decides the particular request is valid or not by using aggregated values of reducers. Alert is generated by the analyzer if there is any attacks happen. The future enhancement of this system is to provide security in all levels in cloud computing.

7. REFERENCES

[1] A Natural Match, "Cloud Computing and Security," April 2010.

[2] Randy Marchany, "Cloud Computing Security Issues," 2010.

[3] "Cloud Computing Making Virtual Machines Cloud-Ready," A Trend Micro White Paper, August 2009.

[4]Frank S. Rietta Duluth, Georgia, "Application Layer Intrusion Detection for SQL Injection," 2006.

[5]Ezumalai R, Aghila G, "Prevention of WebAttacks Using Hirschberg Algorithm".

[6]Jeffrey Dean and Sanjay Ghemawat, "MapReduce: Simplified Data Processing on Large Clusters," OSDI'04.

[7]Tian XIA1,2, "Large-Scale SMS Messages Mining Based on Map-Reduce," Computational Intelligence and Design, 2008.

[8]Wei Wei, Juan Du, Ting Yu, Xiaohui Gu, "SecureMR: A Service Integrity Assurance Framework for MapReduce," Annual Computer Security Applications Conference 2009.

[9]Amir Vahid Dastjerdi, Kamalrulnizam Abu Bakar, Sayed Gholam Hassan Tabatabaei, "Distributed Intrusion Detection in Clouds Using Mobile Agents," ADVCOMP '09.