International Journal of Advances in Scientific Research and Engineering (ijasre)
ISSN: 2454-8006 [Vol. 03, Issue 4, May -2017]
www.ijasre.net

Preventing X-DoS Attack on cloud using Reputation-based Technology

Shruthi R[1] Rasheeda Z Khan[2]

Research Scholar [1,] Professor [2]

Department of Computer Science and Engineering

Shree Devi Institute of Technology

Mangaluru, Karnataka

India

ABSTRACT

Cloud computing is a vast system which provides service based on usage to any number of users from any place using any device. But due to third party interference for maintenance and storage there are numerous denial of service attacks and other security attacks which affects not only the quality but also the cost due to resource consumption. And moreover the cost keeps increasing as long as the DoS attack is not detected at the earliest possible stage. Even though there are many security solutions to keep the attacks away still the DoS attacks take place stealthily due to some point of weakness in the system or the server or application.

The effect of the attack can be so brutalising that it can shutdown an organisation within few seconds and cause billions of damage. Hence special attention has to be given to the stealthy attack. In particular in this paper we are demonstrating X-DoS attack which is one of the most common and dangerous attack that can degrade the system performance stealthily and maximize the financial damage on a cloud user.

And also a method to detect X-DoS attack has been implemented in this paper. The Reputation-Based Technology detects the attack before the attack can actually take place. The RBT checks the reputation of each file before being sent for processing and only files with good reputation are processed thereby preventing X-DoS attack.

Key Words: *X-DoS attack, cloud, Reputation-based Technology.*

---

1. INTRODUCTION

Cloud computing has become so much of our part of our lives as it provides storage, platform and other infrastructure services through a pay as we use method. But the disadvantage is that these services are monitored and quality of these services is regulated by third parties who will lead to certain interference and other possibilities of attacks taking place. Especially the DoS attacks that are taking place even now which cannot be stopped since they are

in a cautious form. These attacks degrade the service and consume more resources making any cloud user or a service vulnerable to attacks.

Many years have been dedicated for the detection of these attacks using the standard approach of finding whether the traffic rate or the flow changes from the normal data flow. But the attackers they know these approach and can easily evade from getting detected by using traffic pattern same as the normal legitimate data and using rapid mutation technology to avoid getting detected.

Here in this paper I have presented a method to show the affects a DDoS attack can make. The impact of the attack is shown by how much the attacker can consume the resources rather than making the service unavailable. Thereby increasing the cost for the cloud user making financial damage. The attack pattern is slowly increased to cause a major damage but without crossing the threshold or getting detected. But if detected a new attack can be initiated.

As in [7] the proposed attack strategy known as the Slowly Escalating Polymorphism DDoS Attack Strategy targets certain vulnerability of the system or the application. It can be applied to various different attacks. The message sequence mutates or changes itself every time a new attack has been initiated in order to avoid detection.

To validate the SIPDAS attack which takes place in a cautious fashion the attack takes place slowly imparting enough damage on the resources which can be shown through graphs and charts. Specifically in this paper I will be demonstrating X-DoS attack which is a serious threat to the cloud system.

These X-DoS attacks can be further stopped using many methods. Like we have the Ingress and Egress Filtering using Access Control lists, using backlogging or back tracking method, using tracing back method, using Netflow,Service Oriented Trace back Architecture or the Reputation Based Technology. In this thesis I have attempted to show how the reputation based technology can stop not the X-DoS attack by checking the fingerprint of every file before being processed.

As in [9] the Reputation-based security technology separates files using various characteristics like age of the page, how many times it has been used, location and more to expose threat. It is built from a global dynamic database which contains data captured from all over the world. RBT is based on advanced data mining techniques and searches for constant changing encryption and mutating code which is used to separate files at danger and from those that are safe.RBT will boost the detection, performance, and accuracy of security system. Hence XML attacks can easily be detected without even being processed at the victim's server or the cloud server. Cyber fraudsters use many techniques to hide themselves like using faster mutation technology to prevent security solutions from identifying the signatures of their code. So the RBT constantly searches for such changing codes and detect them before further damage can be done. Loss in terms of financial cost and service degradation is done making these last for a greater period of time as the attack takes place stealthily.

2. BACKGROUND AND RELATED WORK

In reference [1] it is shown that past years many efforts were devoted for finding detection methods for distributed attacks in network systems as in Fig 1.

The basic standard attack finding methods demonstrate standard ways which were based on data bits per sec, the duration open for the attack and the minimum condition used and differentiated from the real and fake. These methods are already known and hence to hide the signatures and identity for example using rapid mutation technology or zero day attack technology to attack. The fake attack requests are sent in the same slowest way possible like that of the normal information sent which will eventually avoid all reasons for getting under trouble. A standard periodic waveform like the ones used by the low-rate exhausting attacks is not used by these new attacks.

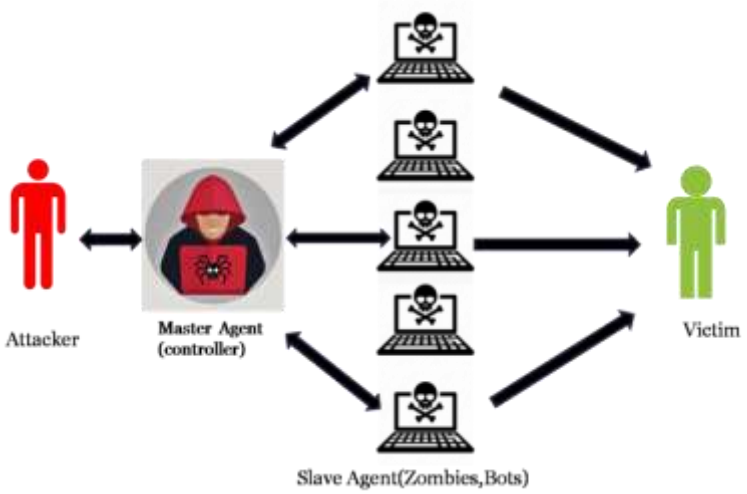

Fig 1 DDoS Architecture

In the paper [2] stealthy word was described which also explains a very slow having symmetric form of data as same as the normal flow of information of a user. These attacks which point to a target will empty and deplete more resources. Since it is a way where no one can find out the consumption will be more and hence damage is more.

In [3] the complicated and more extreme attacks use a certain weak point of a system and make use of that poor design in its attack through which the attacker can easily stop a certain application or descend the performance of the network. The attack can be easily done using a target data job or the scheme. Here the operation which is distributed throughout the system is targeted. The work load will be challenged and its system will be the main culprit.

In [4] LoRDAS attack, the target is the app where its structure is taken into account especially the structure being not easy and that being complex. The similarity between the attack and the user data is so same that these LoRDAS follow a same neutral waveform as that of user sending data. Hence they are very easy to camouflage themselves and hide their presence.

As in [5] in my project I have used the slowly pattern which increases as the attack proceeds. It targets particular application vulnerabilities and then degrades the service. The order of the message is also changed so that no one can detect its pattern.

In [6] to check for the cautious nature of the attack my project design makes use of solutions proposed in the yester work of detecting the cloud attack. The anticipated polymorphic pattern overloads the target system which

causes great amount of financial damage and can avoid or delay from the detection methods. To assess the blow of the attack in a cloud I will be making use of one of grave terrorization to cloud. A Coercive Parsing attack that is X-DoS attack is considered here in my suggested project. The tags are made use of in the attack where they are increased infinitely and sent to cause problems in the opposite side. Since it is infinite in nature the resources used will be continuous and keeps upgrading making expenses really high.

In [7] an X-DoS attack is shown which uses a slowly escalating polymorphic pattern to evade from the detection mechanism. Due to the slowly escalating pattern the attacker is able to impart a maximum loss to a cloud user in terms of financial damage. Here a particular service is just brought down or its speed is decreased impacting on the monetary basis and not on the availability basis. Hence more resources are consumed making a particular usage of a service run longer. A most dangerous attack happening in the cloud computing that is the X-DoS attack is shown and the attack is shown through consumption of resources.

3. CLOUD SYSTEM ANALYSIS

3.1 X-DoS attack System

In this paper I represent an attacking method which is cautious to avoid detections. The attack exploits the cloud flexibility and its resources which forces it to eat up more assets than normal which maximizes fiscal cost and not on the accessibility of service. The attack blueprint makes maximum delay. Like the standard attacks these attacks are opposite in nature to model attacks. The planned attack has a very sluggish rising pattern which will avoid from getting detected. I have also defined an attack detection method which stops the X-DoS attack. This method known as the Reputation Based Technology will check for footprints of a file. If the file has a good reputation in regards to its age, domain, associations, number of users etc it is sent for further processing in the cloud or user server. If the file has a bad reputation it is not processed.

4. X-DOS ATTACK METHODOLOGY.

4.1 Steps for X-DoS Methodology

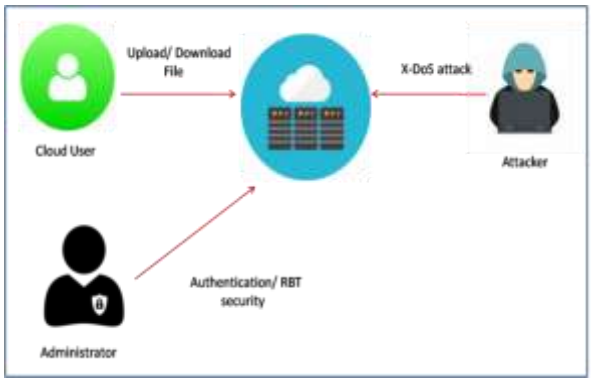

Fig 2.X-DoS Architecture

The following are the steps that are defined in the project for the X-DoS attack as in Fig 2:

1. The cloud user will register and will be authenticated further by the administrator to know if the user is a genuine one or not.
2. The authenticated users can then upload or download a file which will be the legitimate requests that I have considered in my project.
3. These uploaded or downloaded file will then be processed with or without the use of RBT depending on which mode is on.
4. If the RBT security mode is on, the file is checked for its reputation otherwise it is simply sent for processing.
5. The attacker in the same way can send malicious xml file to the cloud user's server and depending on the mode the file is either sent for processing which results in an X-DoS attack.
6. Or the file is sent for RBT security system through which the administrator can stop the attack.
7. If the file has good reputation it is processed otherwise if it has a bad reputation it is not sent for processing.
8. So the reputation of the file saves the admin and the cloud user from getting attacked from any kinds of attack just not only X-DoS attack where the bad files will be listed separate for further action.

4.2. System Architecture and Modular description

The project comprises of the following four modules and three detailed architecture which is described below:

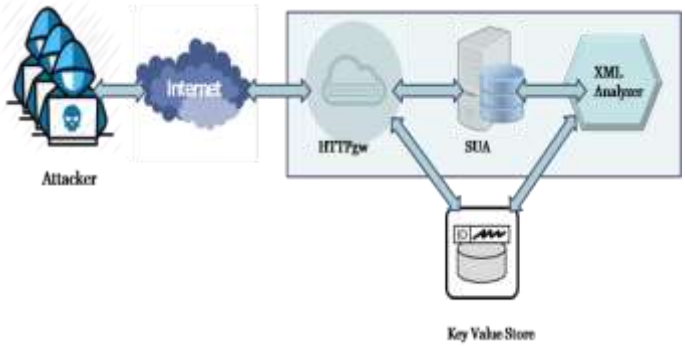

Fig 3.SIPDAS Architecture

Administration Module: The purpose of this module is for user authentication and security of the system.

Master Module: The purpose of this module is for initiation of the X-DoS attack.

Agents Module: This module is used for sending XML message which is a malicious data which starts the X-DoS attack.

Meter Module: The meter performs the necessary requests to the server and evaluates response time.

SIPDAS Architecture

The core elements needed in an X-DoS attack are:

Attacker: The attacker further comprises of a Master, Agents shown in the architecture of the attacker. The attacker initiates the attack with the help of its other components.

HTTPgw: The HTTPgw cloudlet manages the HTTP messages and forwards it to the XML Analyzer as cited in [7].

XML Analyzer: It parses XML document received from the HTTPgw cloudlet and stores the results.

Key-Value store: The results are stored in this store by the XML Analyzer.

SUA: The subsystem for UNIX based application is the target application server. It receives XML messages via HTTP and performs XML parsing and other computations like counting the total no of nested tags.

4.2 Architecture of the attacker

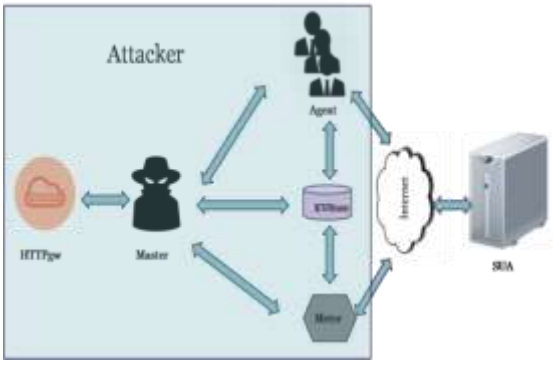

Fig 4. Architecture of the attacker

As shown in Fig 4 when the attack is activated by the attacker a set of parameters like attack intensity attack increment, maximum number of nested tags, threshold are sent to the Master via the Meter. The Master coordinates the attack by interacting with the Agents and the Meter .The Meter performs the requests to the server and evaluates response time for processing the files. The Key-Value store maintains all the information related to the attack state and the attack results evaluated by the Meter. The Master periodically acquires information from the KV store and the Meter and sends message to the Agents in order to update their actions stealthily.

Reputation-Based Technology Architecture

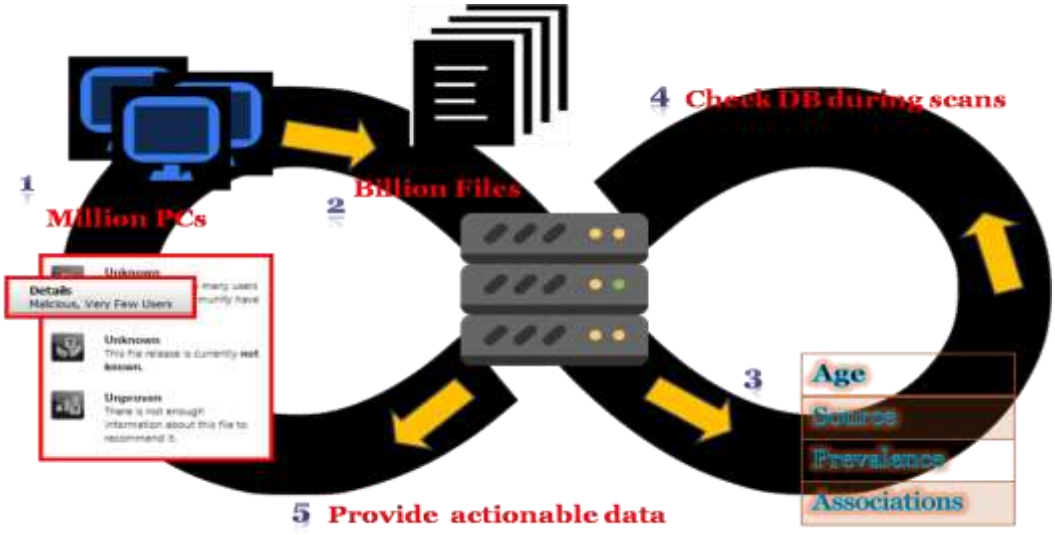

Fig 5. Reputation-Based Technology Architecture

Here in my project I have attempted to simulate a RBT [9] with a dummy database consisting of footprints of both malicious file and legitimate file. All the admin has to do is check for the reputation for a given file before sending for processing. If the file has a good reputation then the administrator allows it for further processing otherwise it's not.
So the steps for Reputation Based Technology in comparison with the real RBT security are as follows:
1. From millions of systems data is collected all over the world and here in the project I have data already collected and stored in the dummy database.
2. So there will be billions of files and the fingerprint of the file saved in a global database using data mining techniques. Here I will be pre-defining fingerprints for all the files the cloud user and the attacker will be sending.
3. So in the real world before the files are processed or downloaded it is scanned for its reputation. Here in my project I will be doing the same thing but I l be showing in two different modes, one is to show the attack and the other to stop the attack.
4. So during the scan the database is checked in both sides.
5. In the end a result will be produced that is to whether the user can process the file or no along with the reasons for such an action.

5. CONCLUSIONS

The project shows the damage an X-DoS attack can make on client system. The damage is so intense that the service is degraded without the user not knowing about it. Here the service is not made unavailable but the service is just degraded to make maximum loss on a client. The xml file that is sent by the attacker to the client slowly exhausts the resources of the client server. This project also shows the counter attack method to prevent the attack from happening. The reputation technology prevents any damage by not allowing the file to be processed until it has a good Reputation.

Many improvements are possible in this project mainly being the simulation which could have been done using a cloud simulator. The project can be designed in a high end perspective using cloudlet and SUA.Data mining techniques can be shown to demonstrate how reputation is saved in databases and some mathematical equations can be made use of in the future.

ACKNOWLEDGMENT

I would like to thank my institution and my guides for their constant support and guidance.

REFERENCES

[1] H. Sun, J. C. S. Lui, and D. K. Yau, "Defending against low-rate TCP attacks: Dynamic detection and protection," in Proc. 12th IEEE Int. Conf. Netw. Protocol., 2004, pp. 196-205.

[2] A. Kuzmanovic and E. W. Knightly, "Low-rate TCP-Targeted denial of service attacks: The shrew vs. the mice and elephants," in Proc. Int. Conf. Appl., Technol., Archit., Protocols Comput. Commun., 2003, pp. 75–86.

[3] M. Guirguis, A. Bestavros, I. Matta, and Y. Zhang, "Reduction of quality (RoQ) attacks on internet end-systems," in Proc. IEEE Int. Conf. Comput. Commun., Mar. 2005, pp. 1362–1372.

[4] X. Xu, X. Guo, and S. Zhu, "A queuing analysis for low-rate DoS attacks against application servers," in Proc. IEEE Int. Conf. Wireless Commun., Netw. Inf. Security, 2010, pp. 500–504.

[5] L. Wang, Z. Li, Y. Chen, Z. Fu, and X. Li, "Thwarting zero-day polymorphic worms with network-level length-based signature generation," IEEE/ACM Trans. Netw., vol. 18, no. 1, pp.53–66, Feb. 2010.

[6] A. Chonka, Y. Xiang, W. Zhou, and A. Bonti, "Cloud security defense to protect cloud computing against HTTP-DOS and XML-DoS attacks," J. Netw. Comput. Appl., vol. 34, no. 4, pp. 1097–1107,Jul. 2011.

[7] Massimo Ficco and Massimiliano Rak," Stealthy Denial of Service Strategy in Cloud Computing," In Proc. IEEE Transactions on Cloud Computing, vol. 3, no 1, January-March 2015.

[8] Digital Attack Map –Top daily DDoS attacks worldwide. (2014) [Online]. Available : http://www.digitalattackmap .com

[9]Symantec INSIGHT -Reputation security technology. (1995-2017)[Online].Available: http://www.symantec.com/reputation-based-securit