A NOVEL METHOD TO MONITOR THREATS IN CLOUD COMPUTING ENVIRONMENT

K.Arthi[1], M.Rajeev Kumar[2], B.Bhagyashree[3]

[1,2]*Associate Professor, Department of CSE, Vel Tech University, Chennai, India*

Abstract *- Cloud Computing did come up with so many attractive advantages such as scalability, flexibility, accessibility, rapid application deployment, and user self-services. However in hindsight, cloud computing makes ensuring security within these environments so much challenging. Therefore traditional security mechanisms such as firewalls and antivirus software have proven insufficient and incapable of dealing with the sheer amount of data and events generated within a Cloud infrastructure. Herein, we present a highly scalable module based system that relies upon Big Data techniques and tools providing a comprehensive solution to process and analyse relevant events (packets flow, logs files) in order to generate an informative decisions that will be handled accordingly and swiftly.[2]As the adoption of Cloud Computing is growing exponentially, a huge sheer amount of data is generated therefore needing to be processed in order to control efficiently what is going within the infrastructure, and also to respond effectively and promptly to security threats. Here in, we provide a highly scalable plug in based and comprehensive solution in order to have a real-time monitoring by reducing the impact of an attack or a particular issue in the overall distributed infrastructure. This work covers a bigger scope in infrastructure security by monitoring all devices that generate log files or generate network traffic. By applying different Big data techniques for Data analysis. [6]*

Keywords (Size 10 & Bold) — *Threats, Cloud computing, SIEM, Data Access*.

I. INTRODUCTION

Due to the evolution of modern computing, the only way to be reliably well informed about a system being compromised is by reviewing the system's actions at both the host and network levels, and then correlating those two levels to develop a thorough view into the system's actions[1]. Also, by the same way, we can upgrade the performance and the capacity if it is altered. In most instances, the end-user often has no indication of the existence of the malicious software or service alteration and therefore cannot be relied upon to determine if their system is indeed compromised or under performance. Without the ability to process the big amount of generated logs in an automated fashion, it would take an administrator days and days to review them and detect the source of a potential problem[3,4]. This method is not recommendable as it does not have the ability to process logs in a near real time especially in a security context which is required to provide actionable intelligence promptly.

This system relies upon centralizing the storage and interpretation of logs and allows near real-time analysis which enables security administrators to take the proper measures more quickly. It also collects data into a central repository for a deep analysis and provides automated reporting for compliance and centralized reporting[5]. SIEM systems work by deploying multiple agents in a hierarchical manner to gather security-related events from end-user devices, servers, network equipment, and even specialized security equipment like firewalls, antivirus and intrusion prevention systems, in order to collect logs and other security-related documentation for analysis. The collection process forwards these events to a centralized management console, which performs rigorous analysis process by inspecting and flagging anomalies. The downside of these approaches is that relevant events may be filtered out too soon; especially that nowadays each data entry is an important ingredient for anomalies detection.

Recently, traditional SIEMs have been overwhelmed by a broad use of log management technologies that focuses on collecting a wide variety of logs from a multitude of sources and for a multitude of purposes and moved to be more persistent and using intelligence, it's called Security Intelligence. From security incident response to regulatory compliance, system management and application troubleshooting, the size of collected data has become bigger and bigger that the issues of storing and managing it instantly is a big challenge.

This paper deals with big data techniques which are used to create value for log data and provides

security for log files which are generated by cloud .In this we can correlate cloud security with big data security.

A. Motivation

Instead of the ability to process the big amount of generated logs in an automated fashion, it would take an administrator days and days to review them and detect the source of a potential problem. This method is not recommendable as it does not have the ability to process logs in a near real time especially in a security context which is required to provide actionable intelligence promptly. From security incident response to regulatory compliance, system management and application troubleshooting, the size of collected data has become bigger and bigger that the issues of storing and managing it instantly is a big challenge. This report deals with big data techniques which are used to create value for log data and provides security for log files which are generated by cloud.

B. Organization of the Paper

The rest of the paper is organized as follows. Part II provides the reader with a background of theory of Big data security, basic information of SIEM and about the cloud as a openstack. Part III provides a detailed review of past literature, while identifying research gaps and scope for further work. Part IV provides all about the work being done by me. It gives the detailed information about the development works and results which I get by implementing in this project. Part V concludes the thesis by summarizing the contributions and indicating a few issues for future work that have been opened up by the studies in this paper.

II. BACKGROUND

A. Big Data Security [2]

Due to the high volume of data generated from different security tools in the form of firewall logs, event logs, application logs, web logs and many other security logs. The efficient handling and processing of this collected data requires high system resources and powerful analysis tools. However, traditional systems and tools are not capable of handling and analysing these large unstructured datasets .In the absence of appropriate processing mechanism for these large datasets, these valuable datasets may become useless and are source overhead for the other important applications.

Security and privacy issues are magnified by velocity, volume and variety of big data, such as large scale cloud infrastructures, diversity of data sources and formats, streaming nature of data acquisition and high volume inter cloud migration. The term big data refers to the massive amounts of digital information companies and governments collect about us and our surroundings. Every day, we create 2.5 quintillion bytes of data—so much that 90% of the data in the world today has been created in the last two years alone. Traditional security mechanisms, which are tailored to securing small-scale static (as opposed to streaming) data, are inadequate[5]. For example, analytics for anomaly detection would generate too many outliers. Similarly, it is not clear how to retrofit provenance in existing cloud infrastructures. Streaming data demands ultra-fast response times from security and privacy solutions.

B. SIEM [4]

SIEM combines security information management (SIM) and security event management (SEM). In both fields the concentration lies on the collection and examination of security relevant data. Though, SEM stresses the combination of data into a convenient amount of info with the help of which safety events can be split with immediately while SIM mainly focuses on the examination of past data in order to develop the long term efficiency and effectiveness of information security infrastructures. The consolidation of SIM and SEM into a combined process of preparation, navigation and monitoring security relevant information on the foundation of data collected from the Information Security (IS) architecture is abridged under the term SIEM. By classifying information and subtracting knowledge from the current capacity of data SIEM struggles to assurance the defence of the information and the information system advantage values of an organization. To accomplish this goal it is essential to conduct SIEM as a combined and incessant organization process[7,8]. In turn, this procedure is reliant on information relevant to judgment making which is mined from the data pool. It is consequently vital to establish suitable practices and mechanisms which provision the operation of data in the supervision process as efficiently and effectively as possible.

By way of a result of the frequent mechanisms connected in an IS architecture, the capacity of protocols as well as the quantity of data generated is huge. Dependent on the system and the accomplishment achieved, log data may cover information about threats or incidences[9].

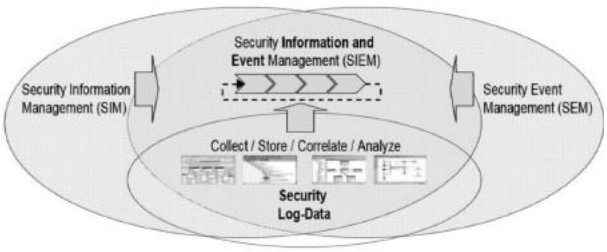


Figure 1: SIEM Architecture

1) Some Valuable event that can be tracked by SIEM:

- The User / administrator activity can be chased (with time and date) and any harms to predefined rules could be informed to the super administrator.
- SIEM solution can show all file access events, particularly the ones in confidential folders.
- All access efforts denied due to admission control limits can be mined by individual operators, etc.
- The log files themselves are observed and variations to them are prompt reported.
- They can monitor all the logs produced by network security paraphernalia's like Firewalls, IDS/IPS, etc. and also the logs of the network gateway devices like routers and correlate all of them for the supervisor to get a better picture of the network activity and to figure out irregular behavior.
- They can observer wired network devices wireless network devices, etc., so that any nameless access to the network can be recognized and ad-hoc variations in User / policies rights can be notified.

SIEM can ensure that anti-virus / OS, Software etc., are present versions, have all up-to-date patches applied and accomplished of generating logs for audit determination. SIEM can analyze which systems in the inner network have been pretentious by malware and are dispersion it to further systems in the network by using correlation methods that look for event designs in multiple systems.

2) *Advantages of SIEM:*

- SIEM help recognize network threats in actual time by capture and study of logs from thousands of devices in numerous branches.
- SIEM allow rapid forensics as they can stock and save all log data from any device for any period.

Figure 2: Advantages of SIEM

3) *Disadvantages of SIEM:*

- SIEM is too complex. Gathering the right data, combining it, regularizing and correlating dissimilar technologies for that one common view is not a minor task.
- SIEM takes too long to deploy. Most establishments looking to participate in a SIEM do so with an intelligence of urgency.
- SIEM is too expensive and too noisy because of it plays alarm many times without correlate the event and threats.

C. *Big Data Security Analytics basically builds on the features that SIEM provides using Hadoop.[5]*

It's building on the process of log collection and putting much more focus on the analysis piece. In order to conduct that analysis, a true security analytics architecture needs to have the ability not only to collect large amounts of log data, but different types of data. Then it needs the infrastructure and analytical firepower to get value out of that data(Here comes Hadoop)It could be defined as a security solution that aggregates and analyses a huge amount of data i.e. very single packet file and flow that is sees on the network. Since we can now view all the data that is flowing across the network we can run our advance intelligence to identify security events from it and Hadoop /Big data is so efficient in managing these huge amount of data it makes our task much more easy. Hadoop can be used to store data from a huge number of systems, such as net flow device, malware analysis systems, routers, Active directory server. The Big Data Security Analytics solutions available today consists of three underlying categories for Advance threat detection:

1) Full security Visibility: The biggest drawback of SIEM was it only used logs and some events to identify attacks, the Big data security solution moves ahead and logs every network packet it seems to be reasonably sure of an attack and also the scope of it For example If a credit card company was attacked and it assumed that the attacker took away 10000 credit card numbers, the SIEM could not say so with authority, since the Big data security solutions sees the exact traffic flowing from the vulnerable server it can pinpoint exactly what was lost. Since the solution captures the exact network packet the analyst can exactly see the web page or the web application as the user saw it.

2) Threat Intelligence: A security solution should remove the guesswork out from uncovering the advance threat by identifying the Malware,Virus, Botnet infected host. For this you need a credible source of information possible from a vendor hosting such information in the cloud, providing services like IP,URL whitelisting and categorization services, or using open source tools like cuckoo, ClamAV, Google safe browsing, robtex.

3) Security Analytics: Attacks classified under Advance persistent attacks are at times hard to detect as the adversary has ample amount of time and lies low for many days in the network before launching

the final assault on your server, detecting these types of attacks are very difficult using traditional SIEM tools. They require more concrete efforts that involve security analytics'. One of the methods that is used to uncover hidden threats is known as relationship mapping, it can help relate the network packets that you saw a week with the one that you are currently seeing and reach conclusions.[2]

D. Cloud as a Openstack

The Openstack community is a global collaboration of developers and cloud computing technologists producing the ubiquitous open source cloud computing platform for public and private clouds. The project aims to deliver feature-rich solutions for all types of clouds by beings simple to implement yet massively scalable. The technology consists of a series of related projects delivering various components for a cloud infrastructure solution. Openstack implements services for establishing infrastructure-as-a-service released under the Apache 2.0 open source license. The project is managed by the Openstack Foundation, an on profit corporate entity established in September 2012 that promotes, protects, and empowers Openstack software and its community. This technology consists of a series of modular projects that control large pools of processing, storage, and networking resources throughout a data center, all managed through a single dashboard that gives administrators control while empowering users to provision resources in a self-service fashion.

Openstack is committed to an open design and development process. The community operates around a six-month, time-based release cycle with frequent development milestones[8].

The Openstack Foundation promotes the development, distribution and adoption of the Openstack cloud operating system. The goal of the Openstack Foundation is to serve developers, users, and the entire ecosystem by providing as setoff shared resources to grow the footprint of public and private Openstack clouds, enable technology vendors targeting the platform and assist developers in producing the best cloud software in the industry.

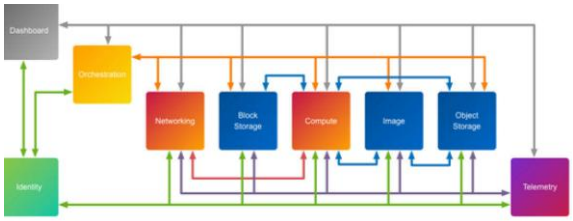


Figure 3: Openstack Architecture

E. Services of Openstack

1) Openstack Compute (project name: Nova) : Openstack enables enterprises and service providers to offer on-demand computing resources, by provisioning and managing large networks of virtual machines. Compute resources are accessible via APIs for developers building cloud applications and through web interfaces for administrators and users. The computer architecture is designed to scale horizontally on standard hardware. Openstack Compute is architected to avoid inherent proprietary hardware or software requirements and the ability to integrate with existing systems and third-party technologies. It is designed to manage and automate pools of compute resources and can work with widely available virtualization technologies, as well as bare metal and high-performance computing configurations[9].

2) Openstack Block Storage (project name: Cinder): Openstack Block Storage provides a "block storage as a service" capability. It provides persistent block devices mapped to Openstack compute instances (which are otherwise assumed to beep humeral).The block storage system manages the creation, attaching and detaching of the block devices to instances. It also optionally supports instance booting and provides mechanisms for creating Snapshot copies and cloning. While fully integrated with Openstack Compute and Dashboard, it can also be used in dependent of Openstack to provide a standardized abstraction for block storage provisioning.

3) Openstack Object Storage (project name: Swift): Openstack Object Storage provides a fully distributed, scale out, API-accessible storage platform that can be integrated directly into applications or used for backup, archiving and data retention. Object storage does not present a traditional file system, but rather a distributed storage system for static data such as virtual machine images, photo storage, email storage, backups and archives. The Openstack Object Storage API (Swift API), in a manner somewhat similar to CDMI, proposes an open standard for cloud storage. It can also function as an alternative endpoint for Amazon Web Services S3 and as a CDMI server through the use of add-on components.

4) Openstack Dashboard (project name: Horizon): The Openstack Dashboard provides administrators and users a graphical interface to access, provision and automate cloud-based resources. The extensible design makes it easy to plug in and expose third-party products and services, such as billing, monitoring, and additional management tools. The dashboard can also be made brand specific for service providers and other Enterprises who require customization[10]. The dashboard is one of several ways to interact with Openstack resources. Developers can automate accessory build tools to manage their resources that use the native Openstack API or the EC2 compatibility API. The dashboard provides users a self-service portal to provision their own resources within the limits set by administrators.

5) Openstack Identity (project name: Keystone): Openstack Identity provides a central directory of users mapped to the Openstack services they can access. It acts as a common authentication system across the cloud operating system and can integrate with existing backend directory services (for example: LDAP). It supports multiple forms of authentication including standard username and password credentials, token based systems and AWS-style logins. Additionally, the catalog provides a list of all of the services deployed in an Openstack cloud in that can be queried in as single registry. Users and third-party tools can programmatically determine which resources they can access[11,12].

Openstack Identity enables:

- Configuration of centralized policies across users and systems.
- Creation of users and tenants and define permissions for compute, storage and networking resources through the use of role based access control (RBAC) features.
- Integration with existing directories, allowing for a single source of identity authentication.
- As a user , get a list of the services that you can access and make API requests or login to the web dashboard to create resources owned by your account

6) Openstack Image Service (project name: Glance): The Openstack Image Service provides discovery, registration and delivery services for disk and server images. The ability to copy or snapshot a server image and immediately store it a way is a powerful capability of the Openstack cloud operating system. Stored images can be used as a template to get new servers up and running quickly and more consistently if you are provisioning multiple servers than installing a server operating system and individually configuring additional services. It can also be used to store and catalogue an unlimited number of backups. The Image Service can store disk and server images in a variety of back-ends, including through NFS and Object Storage. The Image Service API provides a standard REST interface for querying information about disk images and lets clients stream the image stone servers. A multi format image registry allowing uploads of private and public images in a variety of formats[13].

7) Openstack Network Service (project name: Neutron): Openstack Networking is a pluggable, scalable and API-driven system for managing networks and IP addresses. Like other aspects of the cloud operating system, it can be used by administrators and users to increase the value of existing data center assets. Openstack Networking ensures the network is not the bottleneck or limiting factor in a cloud deployment and provides users self-service over their own network configurations. The pluggable backend architecture lets users take advantage of basic commodity gear or advanced networking services from supported vendors. Administrators can take advantage of software defined networking (SDN) technology like Open Flow to allow high levels of multi-tenancy and massive scale. Openstack Networking has an extension framework allowing additional network services, such as intrusion detection systems (IDS), load balancing, firewalls and virtual private networks (VPN) to be deployed and managed.

8) Openstack Telemetry (project name: Ceilometer): Openstack Telemetry provides common infrastructure to collect usage and performance measurements within an Openstack cloud. Its primary initial targets are monitoring and metering, but the framework is expandable to collect data for other needs. Ceilometer was promoted from incubation status to an integrated component of Openstack in the Grizzly (April2013) release.

9) Openstack Orchestration (project name: Heat): Openstack Orchestration implements a service to orchestrate multiple composite cloud applications that use the Amazon Web Services (AWS) Cloud Formation template format, through both an Openstack native and Cloud Formation-compatible API. It is intended, in part, to facilitate movement of workloads from AWS to Openstack deployments. Heat was promoted from incubation status to an integrated component of Openstack in the Grizzly (April 2013) release.

10) Openstack Database as a Service (project name: Trove): Openstack Database as a Service allows users to quickly and easily utilize the features of is rational database without the burden of handling complex administrative tasks. Cloud users and database administrator scan provision and manage multiple database instances as needed. Initially, the service focuses on providing resource isolation high performance while automating complex administrative tasks including deployment, configuration, patching, backups, restores, and monitoring. Trove was promoted from incubation status to an integrated component of Openstack in the Icehouse (April2014) release[14][15].

11) Openstack Hadoop as a Service (project name: Sahara): The Openstack Hadoop as a Service project aims to provide users with simple means to provision a Hadoop cluster by specifying several parameters like Hadoop version, cluster topology, nodes 'hardware details, etc. Sahara was promoted from incubation status to an integrated component of Openstack in the Icehouse (April2014) release.

12) Openstack File Share Service (project name: Manila): Openstack File Share Service provides coordinated access to shared or distributed file systems. While the primary consumption of file shares would be across Openstack Compute instances, the service is also intended to be

accessible as an independent capability in line with the modular design established by other Openstack services. The design and prototype implementation provide extensibility for multiple backends (to support vendor or file system specific nuances/capabilities) but is intended to be sufficiently abstract to accommodate any of a variety of shared or distributed file system types. Manila was officially denoted as an incubated Openstack program during the Juno release cycle.

III.LITERATURE SURVEY

The below are the papers which give abrief idea about big data security and log analysis and big data security application. Each paper shows to deal with the individual issue using a different approach.

Big Data Security Analysis Approach Using Computational Intelligence Techniques in R for Desktop Users is commonly used for the analysis of large volume security data from an organisational perspective, requiring powerful IT infrastructure and expensive data analysis tools. Therefore, it can be considered to be inaccessible to the vast majority of desktop users and is difficult to apply to the rapidly growing data sets for security analysis. A number of commercial companies offer a desktop-oriented big data security analysis solution; however, most of them are prohibitive to ordinary desktop users with respect to cost and IT processing power. This paper presents an intuitive and inexpensive big data security analysis approach using Computational Intelligence (CI) techniques for Windows desktop users, where the combination of Windows batch programming, EmEditor and Rare used for the security analysis. The simulation is performed on a real dataset with more than 10million observations, which are collected from Windows Firewall logs to demonstrate how a desktop user can gain insight in to their abundant and untouched data and extract useful information to prevent their system from current and future security threats. This CI-based big data security analysis approach can also be extended to other types of security logs such as event logs, application logs and weblogs[16].

This paper has presented an intuitive and inexpensive big data security analysis approach using Computational Intelligent (CI) techniques for Windows desktop users. It is based on the combination of Windows batch script, EmEditor (which can be replaced with any powerful editor) and R. This security analysis approach was carried out on a real dataset of 1,006,889,160 bytes (1.01GB) with more than 10 million observations, which were collected in the Windows Firewall log file "pfirewall.log" and integrated in to the "mergedLog" file over the period of 30 days. This desktop-oriented security analysis deduced the security status of the desktop, and sources and causes of the security breaches successfully. Based on the analyses results, a fuzzy inference system was designed to predict the risk of attack and protect the desktop. This security analysis approach and its successful implementation on the modest desktop configuration demonstrate the potential of the proposed approach. However, this particular implementation was limited to the simulated data based on certain firewall rules, a small number of protocols and IP addresses; it would be important to extend rules and areas of investigation, and collect external traffic for making this approach a generalised security analysis approach.

Meta-Analysis of Big Data Security and Privacy collected 79,012 articles from 1916-2016 related to big data to determine which topics were being studied and how much of the literature was focused on privacy or security-related keywords. The analysis demonstrated that the big data paradigm commenced in late 2011and there search production exponentially rose starting in 2012, which approximated we ibull distribution that captured 82% of the variance ($p<.01$). We found there were 13 dominant topics capturing 49% of the big data production in journals during 2011-2016 but privacy and security topics accounted for only 2% and this trend recently dropped to less than 1%. Thus, we argued that we need to stimulate more big data privacy- security research.

This paper analysed 79,012 articles published from 1916 to 2016 related to big data privacy and security. The statistical analysis demonstrated that the big data paradigm commenced in late 2011 and there search publications rose exponentially from 2012 onwards. The search found that there are 13 dominant topics capturing 49% of the big data production in journals during 2011-2016 but privacy and security topics accounted for only 2% in the big data field and this trend recently dropped to less than 1%. The paper, therefore, recommends that more research should be undertaken on big data privacy and security.

Efficient Storage Utilization Using Erasure Codes in Openstack Cloud explains the current era of connected devices, demands of data storage in the cloud storage systems are increasing exponentially and data is being generated at a tremendous speed. To reduce storage overhead, cloud file systems are transitioning from replication to erasure codes. In Openstack, Swift component provides Object storage capabilities. The strategy Swift uses to achieve its reliability, availability and fault - tolerance properties of storage is Replication technique, which keeps more than one copy of each object (typically three). Information Dispersal techniques such as Erasure coding is a method of data protection in which stored information is dispersed in pieces across multiple locations. In this paper we describe this dynamic technique of erasure

code policy in Openstack Swift component to improve to rag efficiency [17].

As a result of this work, we come up with a dynamic technique through implementation of Erasure codes in Openstack Swift component which leads to more utilization of storage and additionally it will also prevents correlated failures from resulting in data loss and mitigates the effect that any single failure has on a storage system or application.

Big Data Security Issues and challenges have entered in data deluge already. Data Deluge means data generated by IoT devices and humans simultaneously. The data deluge is a Big threat for technologist but beneficial for end users. Now the coming problem is the security of this data. Big Data is too big, too fast and too diverse that does not compile with traditional data base system. Traditional data base systems are very good to analyse structured data but these systems are not enough to analyse unstructured data. In this paper we discourse the possible challenges and security issues related to Big Data characteristics and possible solutions.

In this paper, we try to include all the possible challenges and issues related to big data and proposed the solution simultaneously. As for privacy preserving, some data mining techniques like anonymization, randomization etc are already implemented. But as in the real time data, volume, velocity, variety increases at the very high rate so to deal with these challenges simultaneously, modifications in already implemented algorithms are required to develop more reliable and flexible system. The challenges related touser's security and privacy always required the researcher's eye and the previous records and result scan be utilized to extract information for future decision making process. Some more analysis methods must be provided for the anomaly and attack detection that thoroughly depend on the common shared data sets. The era of big data is already begun. At present there are many issues and problems and as the time passes problems will continually arises. Some problems are already solved and some are about to be solved and some solved issues wants further attention to modify the algorithms to maximize the accuracy and speed. Hence, further research is required to develop a robust system.

Big Data Analytics for Security and Privacy Challenges Big Data Analytics for Security intelligence refers to a process of analyzing and mining large amounts of data (petabytes, exabytes, zettabytes) from different sources including IP address, Emails, log files, information get from other attack investigation and many more. Many of the organization use big data analytics for security intelligence to identified anomalies, threat, verify alerts and security events to neutralize cyber-attack. The scope of big data security is limited not only to the current data set brutal so historical data to identified threats, anomalies, and fraud so that network can be safe from targeted attacks. Many institutions are taking steps to focus the growing problems of advanced persistent threats, attacks and fraud. The bigger the better! As more data is collected and retained, the more easily analytics will be able to determine. However companies that use big data analytics must ensure the related privacy and security related issue before implementing analytics into use.

In this paper, we try to summarize some security and privacy related issues that need to focus for constructing big data processing and computing infrastructure extra secure. Security is now a big data problem because the data that has security context is huge. If we are ignoring some of that data or can't analyse it, big data security analyses tools are not security properly. The objective of big data analytics for security is to obtain actionable intelligence in real time. Although big data analytics have significant promises, there are a number of challenges that must be overcome to realize its true potential.

Big data analytics for security and privacy issues focus on the research challenges, leading to greater security and privacy in big data platforms. Following are some of them that need to be addressed:

1. Data provenance
2. Securing big data stores
3. Human computer interaction
4. Privacy
5. Information security as a big data issues.

PROJECT OBJECTIVE

Through this development, we tried to analyse the attacks using SIEM on cloud which is fulfil all parameters of consumers in security. The objective of the thesis is as follows.

"Prevent attacks which are majorly occurring on cloud generated data that is log data. Provide security to that data using SIEM. Create value of semi structured and unstructured data."

The development work presented in this thesis has been aimed to provide security for high end user with optimized features.

IV. IMPLEMENTATION

A. Proposed Model

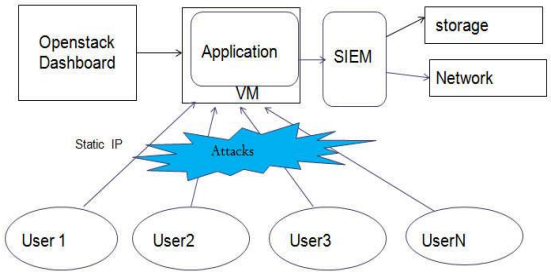


Figure 4: Proposed System

This is the proposed model of our implementation; we created one cloud that is Openstack.
After creating cloud I launch virtual machine in that I develop one login application. There are many N users who login to the application. After login the application, the logs are created that are collected and analyse through the SIEM (Security Information and Event Management).From that we can analyse the attacks.
Processing data would lead us to have information, which would be much more meaningful in a security context. Hence, analyzing and structuring, information would lead to an insightful knowledge that can be put or applied into action (preventing, blocking an attack, etc.)

B. Creating a new image in Openstack Horizon

In the browser, log in to your Openstack Horizon installation. In the left sidebar, under the "project" menu, click image. Create images. In the image creation dialog, fill in the required fields:
Name, Image location, Format
Click on create Image

Figure 5: Openstack dashboard

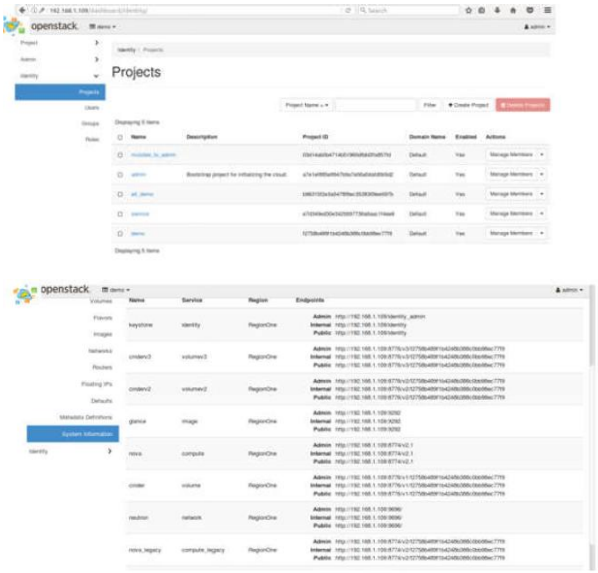


Figure 6: Services of Openstack

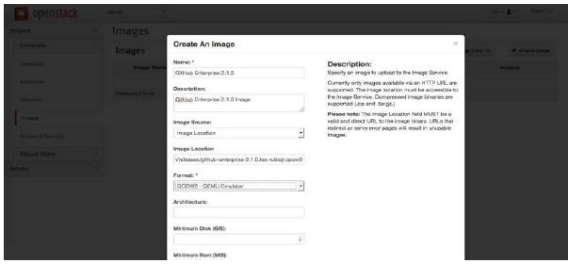


Figure 7: Creation of Image

C. Creating GitHub Enterprise instance

In the left sidebar,click Instances. Click launch instance. In the instance criterion dialog, fill in the required fields:
Instance Name – type a suitable, descriptive name.
Flavor – select a flavor based on your seat count.
Instance Boot Source- select"Boot from Image".
Image Name- selects the name of the image you uploaded in the previous steps.
And then click on Launch.

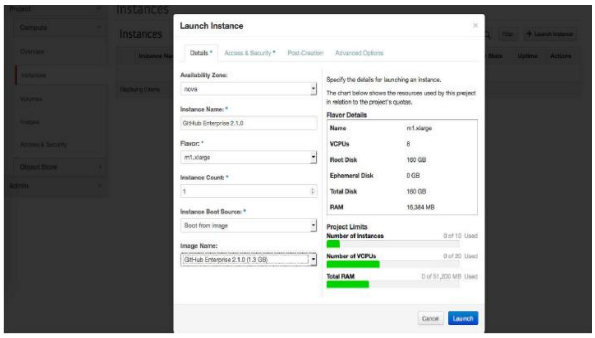


Figure 8: Creation of Instance

D. Creating a security group

If you're setting up your Openstack KVM for the first security, you need to create a security group with entries for each port.
In the left sidebar, click "Access and Security". In the "Security groups" panel, upper right corner, click

create Security group. In this field add necessary information click on create Security Group. To add security rules, in the "Actions" column next to the new security group, click Manage Rules. For each port that needs to add rules.

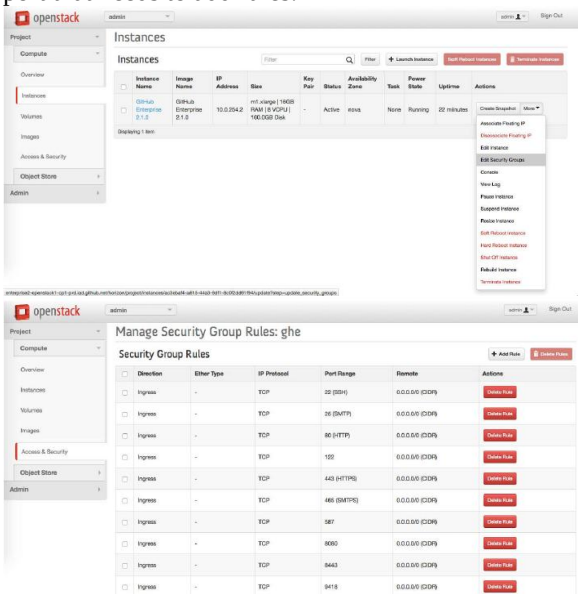


Figure 9: Security to the Instance

E. Adding a storage volume for instance data:

In the left sidebar, click Volumes. Clicks create Volume and fill required fields:
Volume name and size which is in GB. And also manage the volume attainment

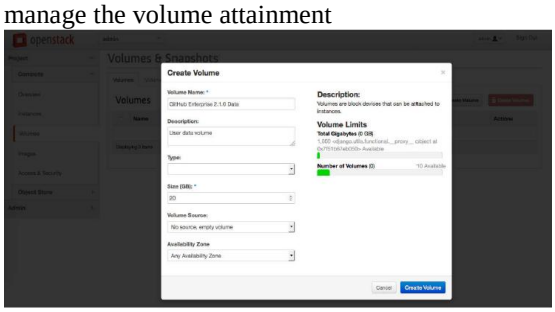
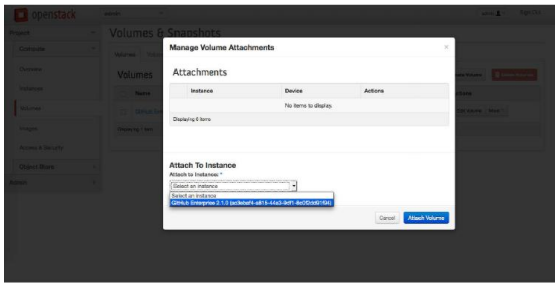


Figure 10: Openstack Volume



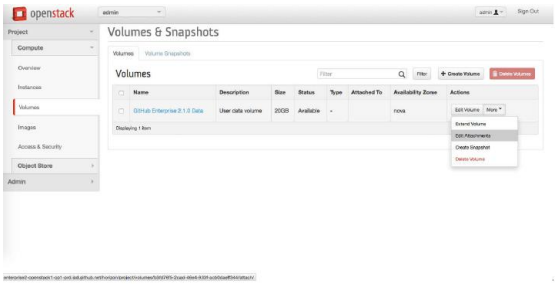


Figure 11: Attaching Volume

After creating and launching all that one login application is develop. If there are N numbers of user who wants to login application then some attacks are occurred that should be examine and analyse through the kibana tool which analyse attacks and prevent.
Some are attacks as follows:
1) Authentication Failures.
2) Authorization Failures.
3) Cross -VM side-channel Attack
4) Denial of Service Attack
5) Brute force Attack

F. Savanna Installation

Figure 12: Savanna instance

Savanna's Architecture

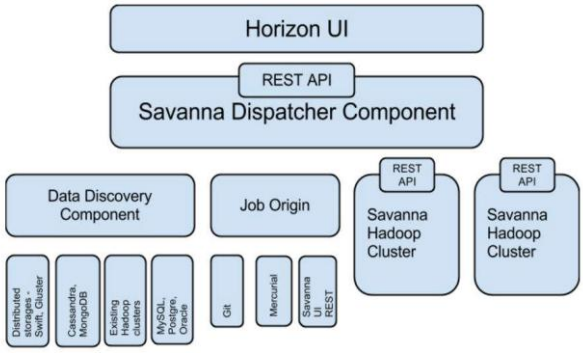


Figure 13: Savanna Architecture

Figure 13 shows the Architecture of savanna:
1) Data Discovery Components – Enable pulling data from various data sources. Data can be pulled from swift, GlusterFs or Nosql database such as cassendra or HBase.
2) Job origin – Supplies a task for data processing and execution on the cluster and allows users to execute several types of job: jar file, pig and hive scripts and oozie job follows.
3) Dispatcher Component – Responsible for Scheduling the job on the new orexisting cluster, provisioning a new cluster, resizing cluster and gathering information from clusters about current jobs and utilization.

4) UI Component – Enables integration with the Openstack (Horizon).It's future intent is to provide instruments for job creation, monitoring and so on. Hue already provides part of this functionality: submitting jobs(jar file, Hive, Pig, Impla), viewing job status and outputting

G. Kibana

Kibana is a web frontend to analyse data held in an elastic search cluster with lot of changes compared to the kibana.

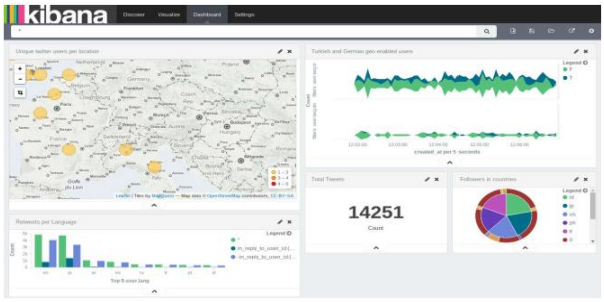


Figure 14: Kibana Dashboard

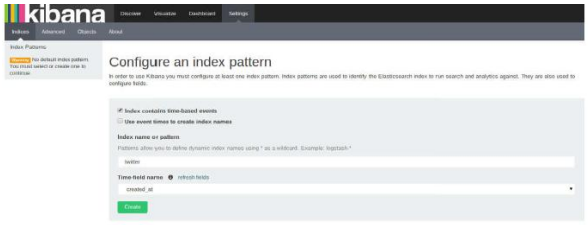


Figure 15: Index pattern

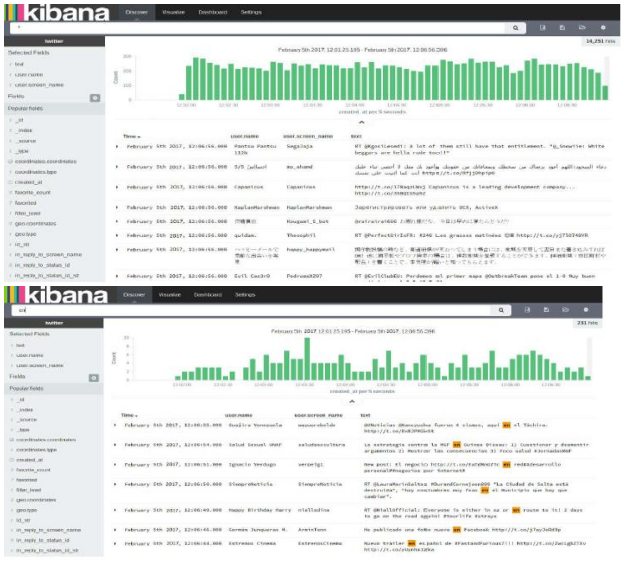


Figure 16: Graphical representation of logs

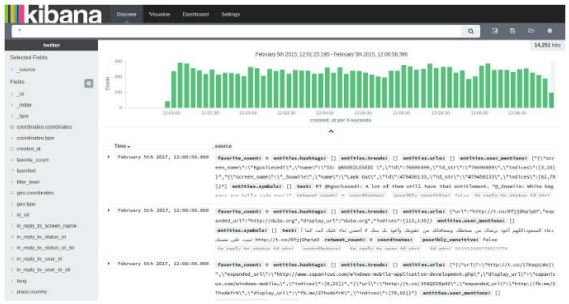


Figure 17: Log Analysis

V. CONCLUSION

There are various algorithms are present to analyse the cloud generated Data. Most efficient was map reduce algorithm/Elasticsearch. To make it work in efficient manner, some big data tools to handle cloud generated data such as log data like kibana. Also various analysing tools need to be used to be used Analyse cloud generated log data which is on large dataset. Provide security and prevent attack by using SIEM.

REFERENCES

[1] Nitin Naik, Paul Jenkins, Nick Savage and Vasilios Katos, "Big Data Security Analysis Approach Using Computational Intelligence Techniques in R for Desktop Users," The Ninth International Conference on Electronic Measurement &Instruments-ICEMI, Beijing, China 2009.

[2] Kenneth David , Zhaohao Sun," Meta-Analysis of Big Data Security and Privacy"@ 2010

[3] Bhagyashri Kulkarni , Varsha Bhosale," Efficient Storage Utilization Using Erasure Codes in OpenStack Cloud," Proc. of IEEE Int. Conf. , Computing and Communication , Durgapur, India 2016.

[4] Neetu Chaudhari, Satyajee Srivastava ,"Big Data Security Issues and challenges" @2015

[5] Aditya Dev Mishra ,Youddha Beer Singh,"Big Data Analytics for Security and Privacy Challenges"@2014

[6] "Toward a Big Data Architecture for Security Events Analytic" Laila Fetjah, Karim Benzidane, Hassan El Alloussi, Othman El Warrak, Said Jai-Andaloussi and Abderrahim Sekkaki @2016 IEEE

[7] "Big Data: Mining of Log File through Hadoop"Bina Kotiyal, Ankit Kumar, Bhaskar Pant, RH Goudar @2015 IEEE

[8] JianwenWEIt*, YusuZHAot, Kaida JIANG*, Rui XIE* and Yaohui , "Analysis Farm: A Cloud-based Scalable Aggregation and Query Platform for Network Log Analysis" JIN*t @ 2011 IEEE.

[9] "Application of Big Data Analytics via Cloud Computing" Yunus Yetis, RuthvikGoud Sara@ 2016 IEEE

[10] The Analysis of Stereo Vision 3D Point Cloud Data of Autonomous vehicle Obstacle Recognition" Li Pei Academy of Armored Forces Engineering, Beijing, China @ 2015 IEEE

[11] https://securityintelligence.com/security-intelligence-and-siem-gets-bigger-with-ease/

[12] http://www.slideshare.net/Hadoop_Summit/t-325p230-ckannanv2

[13] http://www.sciencedirect.com/science/article/pii/S1084804510001281

[14] kibana tutorial

[15] http://www.stackgeek.com/guides/gettingstarted.html

[16] http://www.openstack.com

[17] A. Vijayalakshmi, K. Arthi and P. Vanaja Ranjan, 2013. Network Lifetime Enhancement in Wireless Sensor Networks Using Fuzzy Logic Based Clustering Algorithm. International Journal of Soft Computing, 8: 321-326.