ASSURANCE FRAMEWORK (CPS-CERP) FOR SELECTION OF CLOUD PROVIDER'S SERVICES IN CLOUD ERP

Sunaina Mehta, Dr. Sarvjit Singh Bhatia, Dr.Ashish Oberoi
Research Scholar, Senior Faculty, Associate Professor
RIMT University, GSSDGS, Khalsa College, RIMT University
Mandi Gobindgarh, Patiala, Mandi Gobindgarh , India

Abstract : Cloud computing play an important role in changing the business strategies in competitive market today. Pay per use concept attracted the big or small organizations to adopt cloud computing services. To choose the trustworthy cloud service providers resources in cloud infrastructure based on the requirement of user is a real challenge in cloud. Trust permits the user to select the best resources and the usage of secured services and service providers in commercial cloud environments. Trust value is calculated using three parameters such as Load Time, Data Integrity and Reliability to verify the accuracy of data at client and server side. This paper describes the framework for the storing and retrieving of data on the basis of forming the clusters of different cloud provider's services and selection of the services performing the good quality of secured requirements for the user.

***IndexTerms* - Cloud Computing, Trust, OpenStack, Trust Value**

I. INTRODUCTION

Cloud computing is a model to enable the convenient access to the network request for sharing the groups of configurable calculating resources such as compute, storage etc. Security issues acts as a barrier in the growth of cloud computing. However the cloud environment is attracting great attention to the organizations to progressively switching their data to the cloud platform [1] . Different configuration, resources and services in cloud computing along with various service providers are available in market. Cloud has many features flexibility, elasticity, reliability, availability increased storage, and low cost of computer resources but then major issue of data privacy and security is rising while dealing with the data storage in a cloud. Currently e-commerce, on-line auctioning companies, travel agencies, social networks and other such services use clouds to provide services to their users. Trust is most important factor to be considered for the offered services and applications of cloud between various provider and users. Trust helps the customer in selection of the most reliable and trusted cloud service provider for storing and processing their sensitive information to provide security in cloud computing. Many organizations would like to analyze how to provide the controlled access and authorization, classifying data based on security level with respect to cloud computing environment.

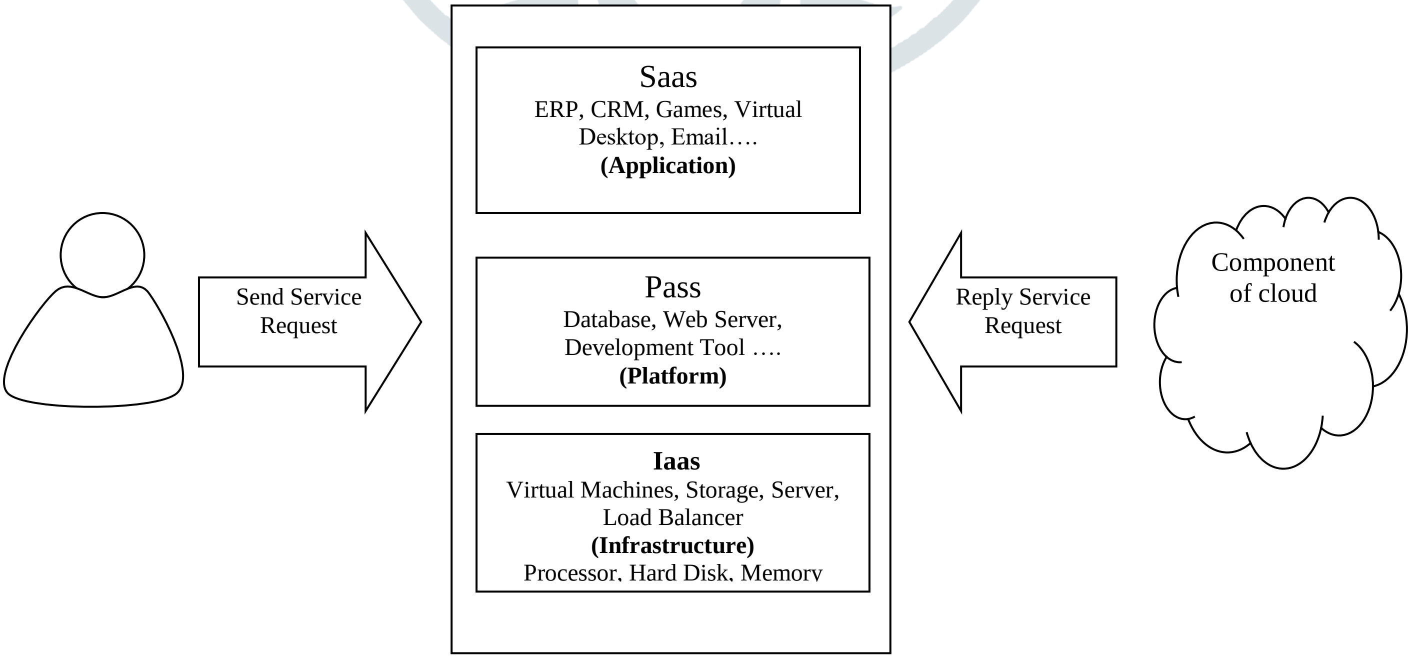

Saas
ERP, CRM, Games, Virtual Desktop, Email….
(Application)
Pass
Database, Web Server, Development Tool ….
(Platform)
Iaas
Virtual Machines, Storage, Server, Load Balancer
(Infrastructure)
Processor, Hard Disk, Memory
Send Service Request
Reply Service Request
Component of cloud

Figure 1: Cloud Computing Environment

In Figure 1, there exist trust relationship between user and cloud service provider. Cloud Service provider (CSP), it provides various cloud service for cloud computing environments, such as Soft ware as a Service, Platform as a Service, Infrastructure as a Service, etc. The user purchases services from the service provider, renting for the computational and storage resources from the infrastructure provider to deploy the virtual instances at minimum cost. Cloud component is the actual carrier of service user is an entity which uses computing resources, storage resources in cloud computing environment. A User sends service request to a cloud service provider then the CSP will reply to the user and provide service.

II. LITERATURE REVIEW

Data Privacy, Security and Trust are the major issues of research in cloud computing. In spite of so much security mechanism provided by the trust for data storage and access still cloud computing is facing the data leakage problems from various attacks. Here we have studied and identified some of the research exist in the literature. Some of the solutions to cloud trust is by highlighting the various methods and designing a framework used as open architecture potential gap between the actual requirement and the real solutions across cloud too [2]. Trust evaluation mechanisms is analyzed and compared them in terms of integrity, security, reliability, dependability, safety, dynamicity, confidentiality, scalability, display systematic literature review (SLR) in the cloud environment [3]. Cloud trust model based on trust level agreement is to provide a hierarchical trust model method to user and improve the security awareness in the cloud computing environments for multi-entities [4] . A questionnaire based approach is the best way to evaluate the security issues in cloud computing. A systematic literature review (SLR) approach on trust evaluation mechanism in cloud computing analyzed the challenges issues of security and trust based solutions [5]. Trust model based on standards of appropriate service quality and speed of implementation for cloud resources is prepared which helps the user to select the appropriate source in cloud infrastructure [6]. Trust mechanism analysis is done on the basis of evidences, attributes certification, validation, service level agreement (SLA), subjective logic and trust is used to evaluate security breaches based on the historical data [7] . Trust model is designed by describing how a service level agreement is prepared combining quality of service requirements of user and capabilities of cloud resource provider [8]. A novel cloud trust model is introduced to solve security issues in which cloud customer can choose different providers’ services and resources in heterogeneous domains in cloud environment [9] . CTrust framework for secure application execution in the cloud system which is scalable to any virtualization software [10]. A trust model measures the security strength and computes a trust value comprises of various parameters helps to ranking services to measure the security in cloud computing [11] .Trust management framework is designed by using of feedback of two parameters SLA, QoS and credibility to calculate trust value by monitoring time and transactions [12] . A secure protocol is derive by analyzing and eliminating the pitfalls of different protocols while accessing data or information along with trust and confidentiality in cloud computing environment [13]. The technology to expose clusters as Web services in the form of Cluster as a Service (CaaS) in cloud using the Resources Via Services (RVWS) framework that quickly select a cluster to specify a job and its execution along with the result file back [14].

We have studied and identified the various risk factors in all aspects of security for application and service in cloud computing. Here we present a proposed framework that can used to evaluate the security strength of the service.

III. NEED OF TRUST IN CLOUD COMPUTING

Data is the valuable asset for any organization. Data privacy and security is major issue while dealing with the data storage in a cloud. To provide the controlled access and authorization, classifying data based on security level criteria becoming area of interest by many organizations using or providing cloud services. Trust is the biggest obstacle for the development of cloud computing. Mutual trust of the users and the services providers are the important factor to evaluate of security issues in cloud computing. Multi tenancy and virtualization are the key features to make efficient utilization of the existing resources and application [15]. A single server, computing facility, data centre and operating system hosts many users, using virtualization. A large number of users are getting served by a cloud provider by this concept of resources sharing. Data protection, communication, resource management for isolation, virtualization etc. are some of the security issues arises due to multi-tenancy and virtualization in the cloud environment. Trust helps the customer in selection of the most reliable and trusted cloud service provider for storing and processing their sensitive information to provide security in cloud computing.

IV. TRUST PARAMETERS

Parameters are useful in measuring the security and checklist of cloud computing environment. Trust value can be a single value is the evaluated by all the individual parameters for checking the security strength of any cloud application or service. Trust parameters are helpful in developing a secure framework in cloud computing environment. The parameters are as below [16].

1. Identity management system (IDM): IDM is a key element for the security of cloud system and for any internet applications. Each user uses his identity for accessing a cloud service. For multiple users in the cloud environment allocation of identities and protecting them requires a strong Identity management system. IDM process includes identity creation, storage and the life cycle management of the identity as various parameters. These processes can be measured against the IDM strength component of the trust model.

2. Authorization: Authorizing is a validation check of user. A user should not be able to use any actions which are not authorized for them. The process of performing for authorized service requires authorization check from provider as well as user side. Various authorization methods like to the stored ACL (Access Control Strength) integrity, Presence of PMI (Privilege Management Information) use by cloud service provides to measure authorization strength.

3. Authentication: To increase the user confidence at login time and identity verification to prevent from unauthorized access for the service, authentication check is required. It is a two way process, for user accessing a service from authentic provider and for provider to give valid services to the user.

4. Data protection and Availability: Data privacy and protection is a major concern while moving data to or from cloud environment. The vital asset of a user as well as any organization moving on to the cloud is data. Data Confidentiality, Integrity and Availability can be measured as the data protection strength.

5. Confidentiality: Confidentiality is required parameter to protect the secrecy of the communication of data and service between a cloud user and provider. Data privacy, message, identity generation are techniques for achieving strength of confidentiality provided by the cloud service.

6. Communication security: When data or messages passed between provider and user in the cloud computing environment prone to eavesdropping or leakage. At the time of data or message transmission lead to measures the communication strength provided by the cloud service. The channel or medium used for communication should be secure and confidential to measures strength of standards for message transmission and communication.

7. Isolation security: Cloud computing multitenant feature leads to the problem of isolation of resources among multiple users. While using the cloud infrastructure and services along with secure isolation of resources is must to increase multiple user confidence in multi-user and multiservice environment.

8. Virtualization security: Virtualization technology introduces other components attacks then physical one. The parameters which lead to measure the security for the protection of application and virtualized resources must be applied in the cloud environment.

V. OPENSTACK

OpenStack is a cloud operating system that enables public and private clouds to be quickly deployed and effectively managed by providing an open-source software service framework that is Application Program Interface (API) driven and pluggable [17] . Fog computing extends the concept of cloud computing functionality and services to network edge by moving computation and storage resources closer to end-users, by reducing the latency and providing new personal services. Fog is a Ruby cloud services library which provides a simplified interface, mocks for testing and powerful features to make cloud easier to work. Fog mocks require no internet connections for testing of data to return. Quality of Service (QoS) and Quality of Experience (QoE) plays important role in the service deployment such as replacing part of a physical smart device with a virtual image to make it remotely usable of the services and related data. In the implementation based on OpenStack, tackles input project by leveraging the fog computing paradigm along with the guarantee of the desired improved levels of security to support the communication among the stakeholders.

VI. CPS-CERP) FRAMEWORK FOR THE SELECTION OF CLOUD PROVIDER'S SERVICES IN CLOUD ERP

Trust shows a big part in profit-making cloud environments. It's solitary of the main contests of cloud technology. Cloud integrity is a wide-ranging term comprises security, privacy, and correctness of data. Trust permits shoppers to selected the best resources in an exceedingly numerous cloud infrastructure.

In Figure 2 describes the (CPS-CERP) framework for ERP in SMEs. User registered themselves on (Cloud-1, Cloud-2, Cloud-3……..). Registration provides Cloud Authentication and Credentials then Create Users and Admin for Upload/Download [18] . Analyze different cloud provider's services by considering three parameters (Load time, Integrity, Reliability) then combine different services of providers and form Clusters. Trust assessment is taken into account by considering three parameters like

1. Load Time Potency

2. Reliability

3. Data Integrity

Optimal dividing of the application mechanism among the device and cloud platforms depends on runtime conditions and trust assessment. Cluster offers their services to a set of nodes. Good performance is conditional and depends on the selection of cluster. Select the cluster and upload file. Download the file and validate the truthfulness of data.

We tried to implement innovative trust model grounded on previous credentials and current talents of a cloud resource supplier. Trust assessment is taken into account by considering three parameters like Turnaround Time potency, Reliability and Data Integrity.

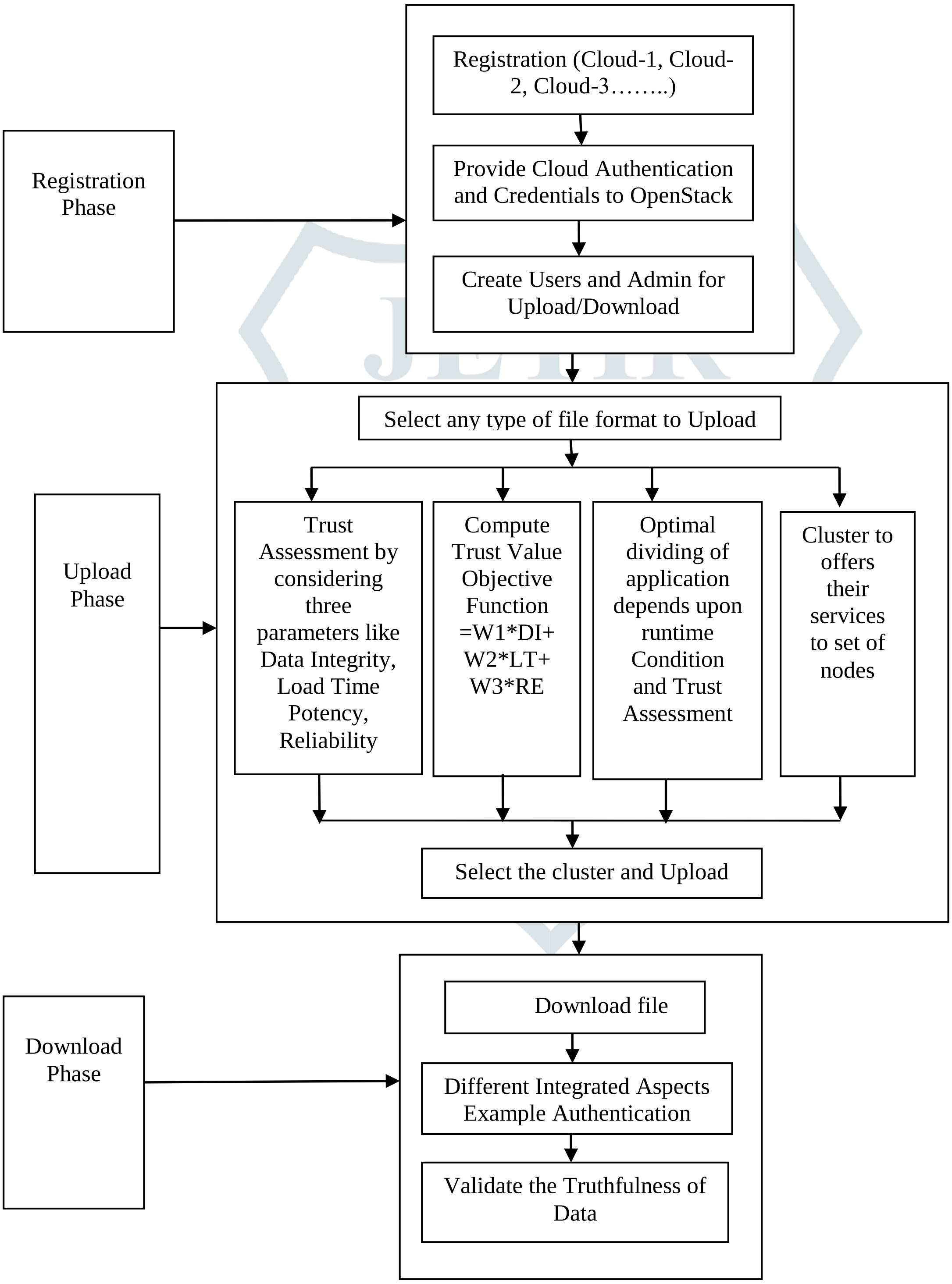

Registration Phase
Registration (Cloud-1, Cloud-2, Cloud-3……..)
Provide Cloud Authentication and Credentials to OpenStack
Create Users and Admin for Upload/Download
Upload Phase
Select any type of file format to Upload
Trust Assessment by considering three parameters like Data Integrity, Load Time Potency, Reliability
Compute Trust Value Objective Function =W1*DI+ W2*LT+ W3*RE
Optimal dividing of application depends upon runtime Condition and Trust Assessment
Cluster to offers their services to set of nodes
Select the cluster and Upload
Download Phase
Download file
Different Integrated Aspects Example Authentication
Validate the Truthfulness of Data

Figure 2: (CPS-CERP) Framework for selection of cloud Provider's services in cloud ERP

VII. Analysis of Cloud Provider's Services:

1. Load Time

Load Time -Time units between the submission of a job and the delivery of the completed job

LT_{time} = Td-Ts
T_d: Delivery Time
T_s: Submission Time

2. Data Integrity

Data integrity is a comprehensive term and it comprises security, confidentiality and accurateness of the data. How to choose a cloud resource specified its trust and capabilities?

3. Reliability of a cloud resource

It is a measure of successful completion of accepted jobs by the cloud resource. Out of A_k jobs accepted by resource R_k, let C_k denote the number of jobs completed successfully by resource R_k over the period T.

Reliability of a cloud resource= jobs completed successfully ($C_{k)}$ /jobs accepted by resource (A_k)

The QoS has the following factors:

1. Computing Power - CPU, RAM, Hard disk capacity.

2 . Data Integrity

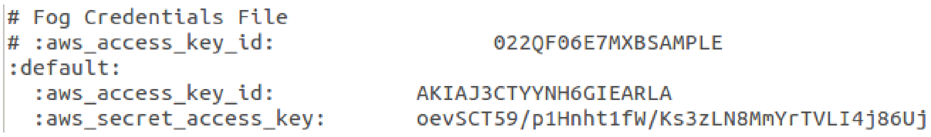
In a heterogeneous surroundings, this belief of trust is certainly problematic to accurately tell, so there is no completely recognized description of trust in cloud computing. However, by reducing, removing, and/or allocating trust relations among cloud structure constituents and customers, one can create comparative, incremental enhancements to the trustworthiness of clouds, refining their safety.

The server stores encrypted data and it is decrypted at client side. Data is secure at server and third party cannot access the data. Concern is disloyal behaviour of the cloud. The data stored on cloud may be transformed without the awareness of client. Therefore a mechanism that verifies the stored should be present at cloud to verify data being retrieved at client is same [19] . At client side after decrypting the data certain rules can be set to verify uniqueness of the data. This verifies the data has been transformed or it is same as the original data put in storage by the client.

Trust Value of a Cloud Provider =W1*DI+W2*LT+W3*RE.

Where w1, w2 and w3 are positive weights of the trust parameters such that w1+w2+w3 = 1. The weights of the trust attributes are predetermined based on their priority. For example, w1 = 0.7, w2 = 0.1, w3=0.2, Here data integrity is given the highest priority whereas load time efficiency is given the lowest priority.

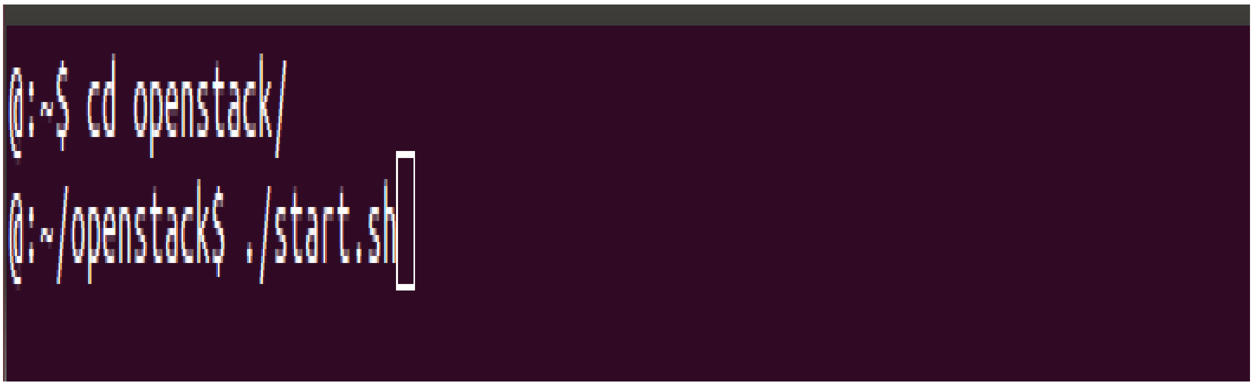
Initial step is to Setup Credential file as shown in figure 3



```
# Fog Credentials File
# :aws_access_key_id:                 022QF06E7MXBSAMPLE
:default:
  :aws_access_key_id:           AKIAJ3CTYYNH6GIEARLA
  :aws_secret_access_key:       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
```

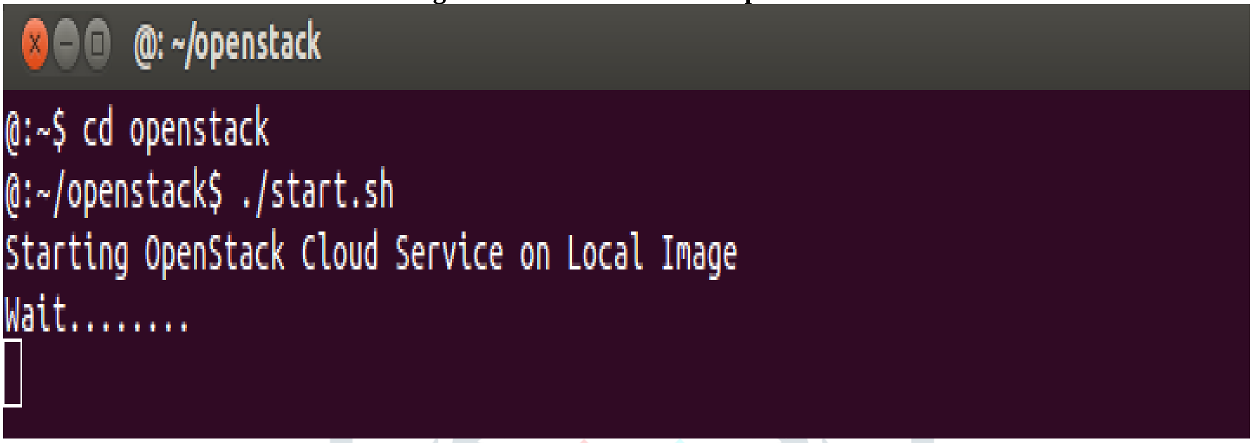
Figure 3: Fog Credentials File

OpenStack software controls large pools of compute, storage, and networking resources. OpenStack services are organized following the Shared Nothing principle. Each instance of a service (i.e., service worker) is exposed through an API accessible through a Remote Procedure Call (RPC). One of the strengths of OpenStack is that it exposes a very rich API that can be used to control every aspect of your cloud.



```
@:~$ cd openstack/
@:~/openstack$ ./start.sh
```

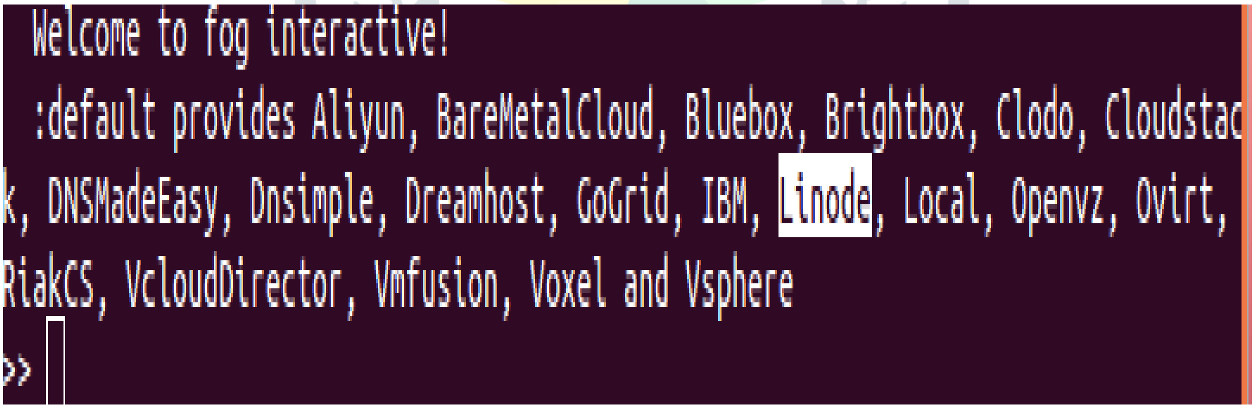
Figure 4: Command to Star openstack



```
@:~$ cd openstack
@:~/openstack$ ./start.sh
Starting OpenStack Cloud Service on Local Image
Wait........
```

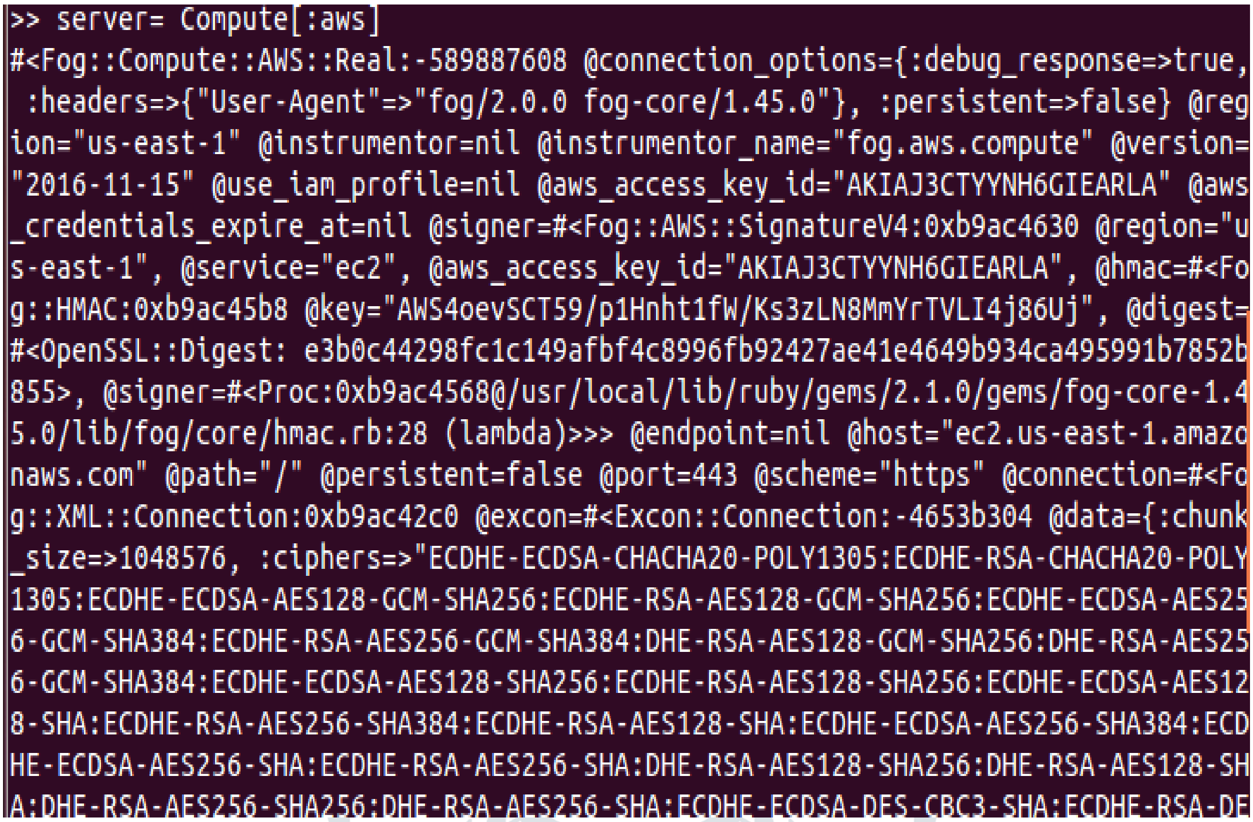
Figure 5: Starting openstack

One of the more interesting ways of interacting with an Openstack cloud is programmatically. There is a Ruby Gem named Fog that allows such interaction.



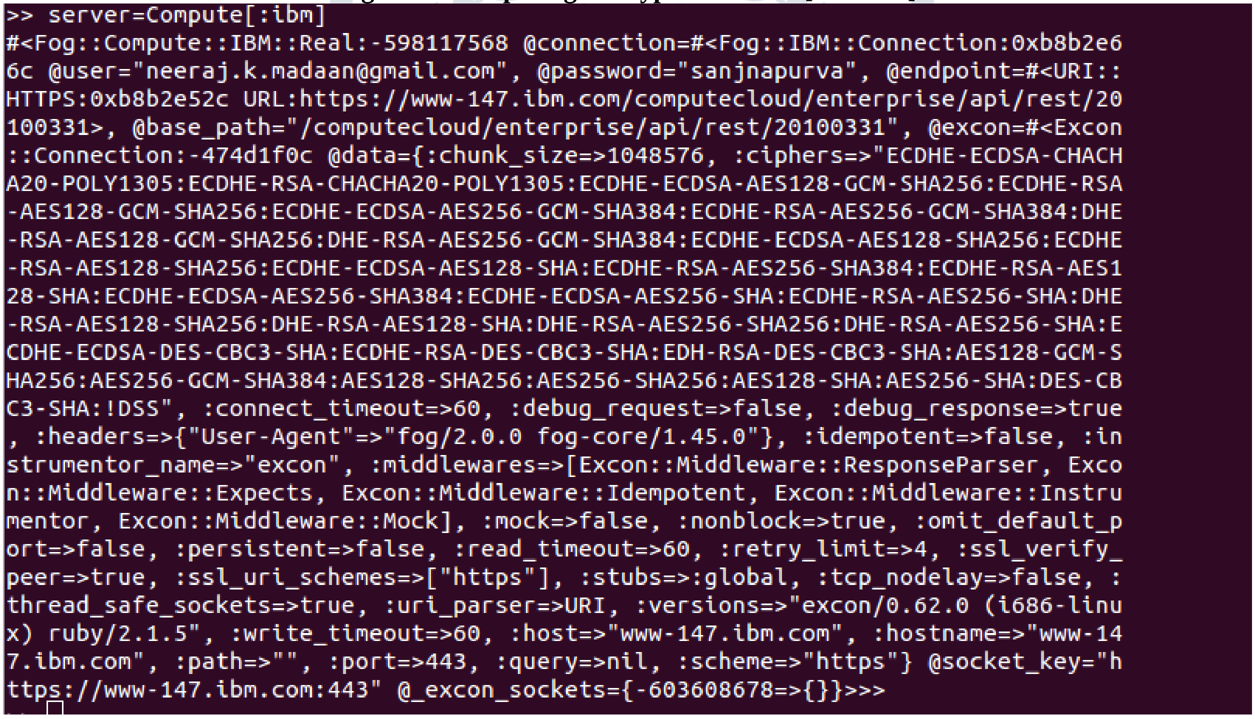
```
  Welcome to fog interactive!
  :default provides Aliyun, BareMetalCloud, Bluebox, Brightbox, Clodo, Cloudstac
k, DNSMadeEasy, Dnsimple, Dreamhost, GoGrid, IBM, Linode, Local, Openvz, Ovirt,
RiakCS, VcloudDirector, Vmfusion, Voxel and Vsphere
>>
```

Figure 6: Default Providers



```
>> server= Compute[:aws]
#<Fog::Compute::AWS::Real:-589887608 @connection_options={:debug_response=>true,
 :headers=>{"User-Agent"=>"fog/2.0.0 fog-core/1.45.0"}, :persistent=>false} @reg
ion="us-east-1" @instrumentor=nil @instrumentor_name="fog.aws.compute" @version=
"2016-11-15" @use_iam_profile=nil @aws_access_key_id="AKIAJ3CTYYNH6GIEARLA" @aws
_credentials_expire_at=nil @signer=#<Fog::AWS::SignatureV4:0xb9ac4630 @region="u
s-east-1", @service="ec2", @aws_access_key_id="AKIAJ3CTYYNH6GIEARLA", @hmac=#<Fo
g::HMAC:0xb9ac45b8 @key="AWS4oevSCT59/p1Hnht1fW/Ks3zLN8MmYrTVLI4j86Uj", @digest=
#<OpenSSL::Digest: e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b
855>, @signer=#<Proc:0xb9ac4568@/usr/local/lib/ruby/gems/2.1.0/gems/fog-core-1.4
5.0/lib/fog/core/hmac.rb:28 (lambda)>>> @endpoint=nil @host="ec2.us-east-1.amazo
naws.com" @path="/" @persistent=false @port=443 @scheme="https" @connection=#<Fo
g::XML::Connection:0xb9ac42c0 @excon=#<Excon::Connection:-4653b304 @data={:chunk
_size=>1048576, :ciphers=>"ECDHE-ECDSA-CHACHA20-POLY1305:ECDHE-RSA-CHACHA20-POLY
1305:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES25
6-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES25
6-GCM-SHA384:ECDHE-ECDSA-AES128-SHA256:ECDHE-RSA-AES128-SHA256:ECDHE-ECDSA-AES12
8-SHA:ECDHE-RSA-AES256-SHA384:ECDHE-RSA-AES128-SHA:ECDHE-ECDSA-AES256-SHA384:ECD
HE-ECDSA-AES256-SHA:ECDHE-RSA-AES256-SHA:DHE-RSA-AES128-SHA256:DHE-RSA-AES128-SH
A:DHE-RSA-AES256-SHA256:DHE-RSA-AES256-SHA:ECDHE-ECDSA-DES-CBC3-SHA:ECDHE-RSA-DE
```

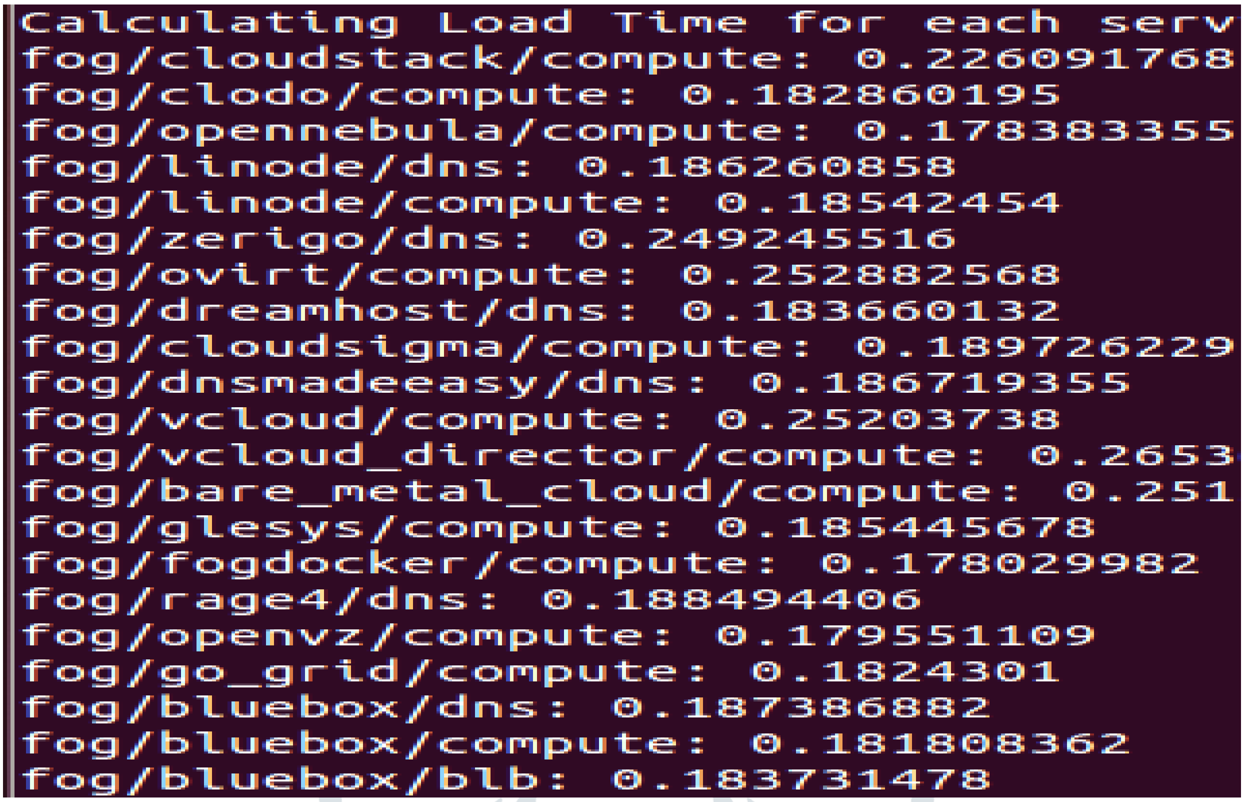
Figure 7: Computing Encryption Details [Amazon]



```
>> server=Compute[:ibm]
#<Fog::Compute::IBM::Real:-598117568 @connection=#<Fog::IBM::Connection:0xb8b2e6
6c @user="neeraj.k.madaan@gmail.com", @password="sanjnapurva", @endpoint=#<URI::
HTTPS:0xb8b2e52c URL:https://www-147.ibm.com/computecloud/enterprise/api/rest/20
100331>, @base_path="/computecloud/enterprise/api/rest/20100331", @excon=#<Excon
::Connection:-474d1f0c @data={:chunk_size=>1048576, :ciphers=>"ECDHE-ECDSA-CHACH
A20-POLY1305:ECDHE-RSA-CHACHA20-POLY1305:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA
-AES128-GCM-SHA256:ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:DHE
-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES128-SHA256:ECDHE
-RSA-AES128-SHA256:ECDHE-ECDSA-AES128-SHA:ECDHE-RSA-AES256-SHA384:ECDHE-RSA-AES1
28-SHA:ECDHE-ECDSA-AES256-SHA384:ECDHE-ECDSA-AES256-SHA:ECDHE-RSA-AES256-SHA:DHE
-RSA-AES128-SHA256:DHE-RSA-AES128-SHA:DHE-RSA-AES256-SHA256:DHE-RSA-AES256-SHA:E
CDHE-ECDSA-DES-CBC3-SHA:ECDHE-RSA-DES-CBC3-SHA:EDH-RSA-DES-CBC3-SHA:AES128-GCM-S
HA256:AES256-GCM-SHA384:AES128-SHA256:AES256-SHA256:AES128-SHA:AES256-SHA:DES-CB
C3-SHA:!DSS", :connect_timeout=>60, :debug_request=>false, :debug_response=>true
, :headers=>{"User-Agent"=>"fog/2.0.0 fog-core/1.45.0"}, :idempotent=>false, :in
strumentor_name=>"excon", :middlewares=>[Excon::Middleware::ResponseParser, Exco
n::Middleware::Expects, Excon::Middleware::Idempotent, Excon::Middleware::Instru
mentor, Excon::Middleware::Mock], :mock=>false, :nonblock=>true, :omit_default_p
ort=>false, :persistent=>false, :read_timeout=>60, :retry_limit=>4, :ssl_verify_
peer=>true, :ssl_uri_schemes=>["https"], :stubs=>:global, :tcp_nodelay=>false, :
thread_safe_sockets=>true, :uri_parser=>URI, :versions=>"excon/0.62.0 (i686-linu
x) ruby/2.1.5", :write_timeout=>60, :host=>"www-147.ibm.com", :hostname=>"www-14
7.ibm.com", :path=>"", :port=>443, :query=>nil, :scheme=>"https"} @socket_key="h
ttps://www-147.ibm.com:443" @_excon_sockets={-603608678=>{}}>>>
```

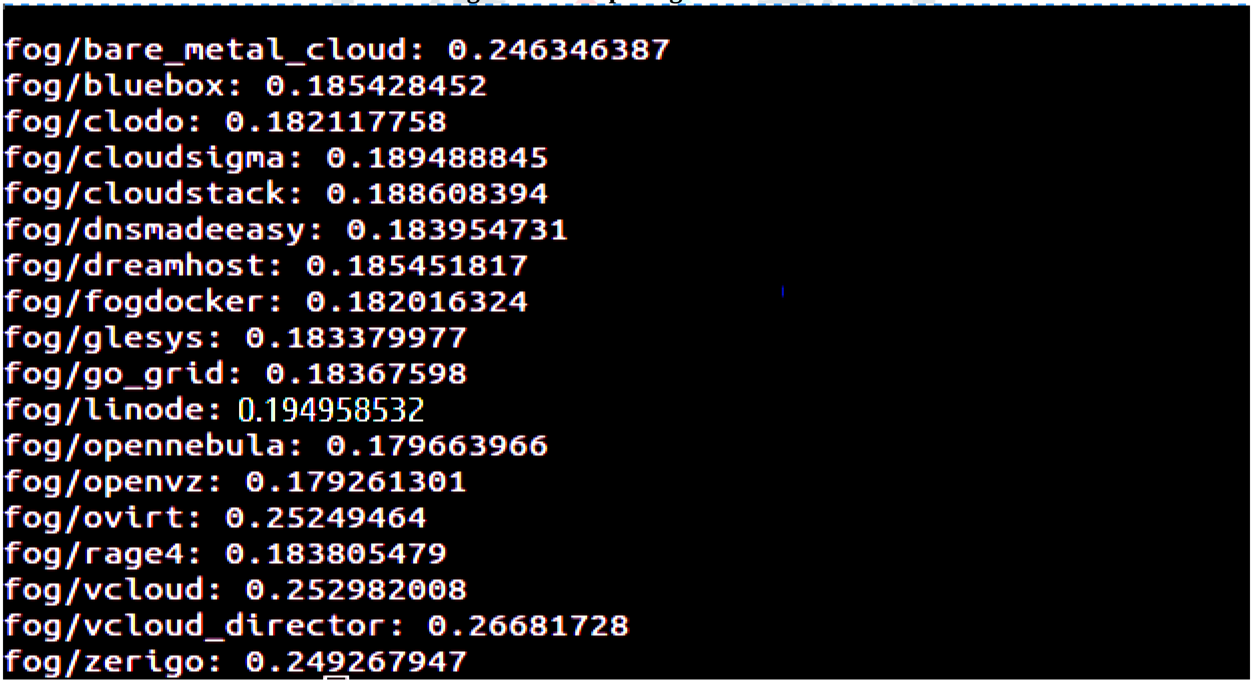
Figure 8: Computing Encryption Details [IBM]

Computing Trust



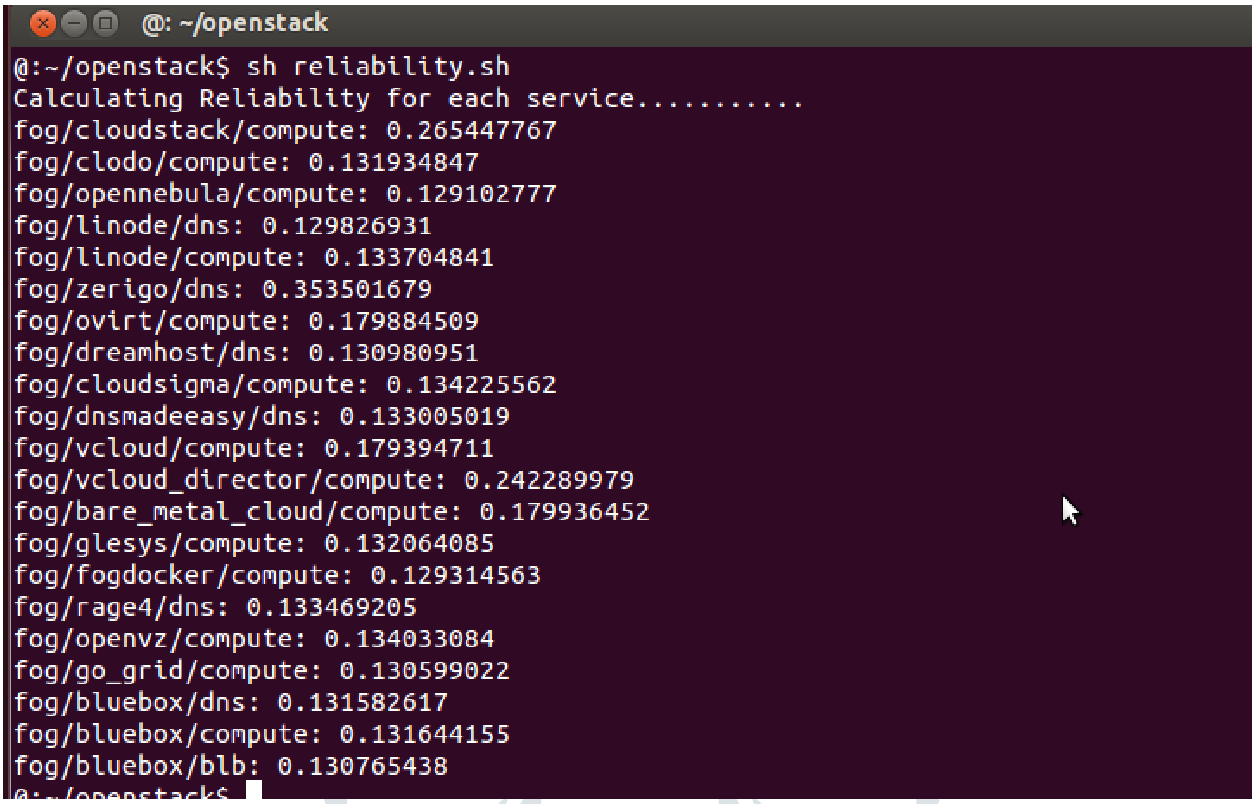
```
Calculating Load Time for each serv
fog/cloudstack/compute: 0.226091768
fog/clodo/compute: 0.182860195
fog/opennebula/compute: 0.178383355
fog/linode/dns: 0.186260858
fog/linode/compute: 0.18542454
fog/zerigo/dns: 0.249245516
fog/ovirt/compute: 0.252882568
fog/dreamhost/dns: 0.183660132
fog/cloudsigma/compute: 0.189726229
fog/dnsmadeeasy/dns: 0.186719355
fog/vcloud/compute: 0.25203738
fog/vcloud_director/compute: 0.2653
fog/bare_metal_cloud/compute: 0.251
fog/glesys/compute: 0.185445678
fog/fogdocker/compute: 0.178029982
fog/rage4/dns: 0.188494406
fog/openvz/compute: 0.179551109
fog/go_grid/compute: 0.1824301
fog/bluebox/dns: 0.187386882
fog/bluebox/compute: 0.181808362
fog/bluebox/blb: 0.183731478
```

Figure 9: Computing Load Time



```
fog/bare_metal_cloud: 0.246346387
fog/bluebox: 0.185428452
fog/clodo: 0.182117758
fog/cloudsigma: 0.189488845
fog/cloudstack: 0.188608394
fog/dnsmadeeasy: 0.183954731
fog/dreamhost: 0.185451817
fog/fogdocker: 0.182016324
fog/glesys: 0.183379977
fog/go_grid: 0.18367598
fog/linode: 0.194958532
fog/opennebula: 0.179663966
fog/openvz: 0.179261301
fog/ovirt: 0.25249464
fog/rage4: 0.183805479
fog/vcloud: 0.252982008
fog/vcloud_director: 0.26681728
fog/zerigo: 0.249267947
```

Figure 10: Computing Integrity



```
@:~/openstack$ sh reliability.sh
Calculating Reliability for each service...........
fog/cloudstack/compute: 0.265447767
fog/clodo/compute: 0.131934847
fog/opennebula/compute: 0.129102777
fog/linode/dns: 0.129826931
fog/linode/compute: 0.133704841
fog/zerigo/dns: 0.353501679
fog/ovirt/compute: 0.179884509
fog/dreamhost/dns: 0.130980951
fog/cloudsigma/compute: 0.134225562
fog/dnsmadeeasy/dns: 0.133005019
fog/vcloud/compute: 0.179394711
fog/vcloud_director/compute: 0.242289979
fog/bare_metal_cloud/compute: 0.179936452
fog/glesys/compute: 0.132064085
fog/fogdocker/compute: 0.129314563
fog/rage4/dns: 0.133469205
fog/openvz/compute: 0.134033084
fog/go_grid/compute: 0.130599022
fog/bluebox/dns: 0.131582617
fog/bluebox/compute: 0.131644155
fog/bluebox/blb: 0.130765438
@:~/openstack$
```

Figure 11: Computing Reliability

VIII. Computation of Load Time, Data Integrity and Reliability

Fog Default Provider	Load time for Service(Compute)	Data Integrity For Each Provider	Reliability
BareMetalCloud	0.251588154	0.246346387	0.179936452
Clodo	0.182860195	0.182117758	0.131934847
CloudSigma	0.189726229	0.189488845	0.134225562
Cloudstack	0.226091768	0.188608394	0.265447767
DNSMadeEasy	0.186719355	0.183954731	0.133005019
Dreamhost	0.183660132	0.185451817	0.130980951
Fogdocker	0.178029982	0.182016324	0.129314563
GoGrid	0.1824301	0.18367598	0.130599022
Glesys	0.185445678	0.183379977	0.132064085
Linode	0.18542454	0.194958532	0.133704841
OpenNebula	0.178383355	0.179662966	0.129102777
OpenVZ	0.179551109	0.179261301	0.134033084
Ovirt	0.252882568	0.25249464	0.179884509
Vcloud	0.25203738	0.252982008	0.179394711
VcloudDirector	0.265361099	0.26681728	0.242289979
Zerigo	0.249245516	0.249267947	0.353501679

Table 1: Load Time, Data Integrity and Reliability Computation

IX. CONCLUSION

To store a secured, integrated, private data along with minimum cost in cloud computing is the major issues for the SMEs. Trust is the conformation of several elements such as consistency, honesty, reliability, steadiness, security, capability. This paper presents an analysis of trust solution by combining different cloud providers services in the form the clusters in cloud environment. The framework was implemented as a working module by selecting cluster of cloud service providers for optimal dividing of application among device for speedy accurate performanance of data. The trust value measures the accuracy of stored data in remote cloud.

In Openstack software Fog Default Providers are setup and load time for service, Data Integrity for each provider and Reliability is computed in cloud with Ruby Gem Fog. A different service of cloud providers combine to form the cluster for the communication on internet. For the future, we aim to implement a framework in selection of the best cloud services provided by the cloud service providers to achieve good performanance by optimal dividing of the application among devices.

REFERENCES

[1] Chee Shin Yeo, Srikumar Venugopal,James Broberg , Ivona Brandic , Rajkumar Buyya, "Cloud computing and emerging IT platforms: Vision, hype, and reality fordelivering computing as the 5th utility," *Future Generation Computer Systems, Elsevier*, pp. 1-19, June 2009.

[2] Gokulnath K. and Rhymend Uthariaraj, "A Survey on Trust Models in Cloud Computing," *Indian Journal of Science and Technology*, vol. 9, no. 47, pp. 1-7, December 2016.

[3] Nima Jafari Navimipour, Matin Chiregi, "Cloud computing and trust evaluation: A systematic literature reviewof the state-of-the-art mechanisms," *Journal of Electrical Systems and Information Technology*, pp. 1-15, September 2017.

[4] Wu, Xu, "Study on Trust Model for Multi-users in Cloud Computing," *International Journal of Network Security* , vol. 20, no. 4, pp. 674-682, July 2018.

[5] Nima Jafari Navimipour, Matin Chiregi, "Cloud computing and trust evaluation: A systematic literature review of the state-of-the-art mechanisms," *Journal of Electrical Systems and Information Technology*, pp. 1-15, September 2017.

[6] Atoosa Gholami and Mostafa Ghobaei Arani, "A Trust Model Based on Quality of Service in Cloud Computing Environment," *International Journal of Database Theory and Application*, vol. 8, no. 5, pp. 161-170, Nov 2015.

[7] Jingwei Huang and David M Nicol, "Trust mechanisms for cloud computing,Springer," *Journal of Cloud Computing*, vol. 9, pp. 1-14, 2013.

[8] Paul Manuel, "A trust model of cloud computing based on Quality of Service," *Annals of Operations Research, Springer*, pp. 1-12, April 2013.

[9] Wenjuan Li and Lingdi Ping, "Trust Model to Enhance Security and Interoperability of Cloud Environment," *CloudCom, Springer*, pp. 69-79, 2009.

[10] Srujan Kotikela,Mahadevan Gomathisankaran, Satyajeet Nimgaonkar, "CTrust: A framework for Secure and Trustworthy application execution in Cloud computing," in *International Conference on Cyber Security (IEEE Computer Society)*, 2012, pp. 24-31.

[11] M. Sasikumar, Rizwana Shaikh, "Trust Model for Measuring Security Strength of Cloud Computing Service," *Procedia Computer Science , Elsevier*, vol. 45, pp. 380 – 389, 2015.

[12] Smriti Kumar Sinha, Monoj Kumar Muchahari, "A New Trust Management Architecture for Cloud Computing Environment," in *International Symposium on Cloud and Services Computing ,IEEE*, 2012, pp. 136-140.

[13] Yang Xiang,Shawkat Ali, Mahbub Ahmed, "Above the Trust and Security in Cloud Computing:A Notion towards Innovation," in *International Conference on Embedded and Ubiquitous Computing, IEEE/IFIP*, Australia, 2010, pp. 723-730.

[14] Michael Brock and Andrzej Goscinski, A Technology to Expose a Cluster as a Service in a Cloud, 2010, Proc.,8th Australasian Symposium on Parallel and Distributed Computing (AusPDC), Brisbane, Australia,2010.

[15] M. Sasikumar, Rizwana Shaikh, "Security Issues in Cloud Computing: A survey," *International Journal of Computer*

Applications, vol. 44, no. 19, pp. 4-10, April 2012.

[16] M.Sasikumar, Rizwana A.R.Shaikh, "Trust Model for a Cloud Computing Application and Service," in *International Conference on Computational Intelligence and Computing Research IEEE*, 2012, pp. 1-4.

[17] G. Genovese,A. Iera,P.Lago,G. Lamanna,C. Lombardo,S. Mangialardi, R. Bruschi, "OpenStack Extension for Fog-Powered Personal Services Deployment," in *29th International Teletraffic Congress, ITC*, 2017, pp. 19-23.

[18] Costas Lambrinoudakis, Zafeiroula Georgiopoulou, "Trust Managemement Parameters in Cloud Computing Environments," in *The Eighth International Conference on Cloud Computing, GRIDs, and Virtualization*, 2017, pp. 152-155.

[19] Supriya Kinger, Randeep Kaur, "Analysis of Security Algorithms in Cloud Computing," *International Journal of Application or Innovation in Engineering & Management (IJAIEM)*, vol. 3, no. 3, pp. 171-176, March 2014.