SERVICE LEVEL AGREEMENT CHECKING IN CLOUD COMPUTING IN TERMS OF VERIFICATION AND VALIDATION CONCEPTS

Nor Shahida Mohd Jamail, Rodziah Atan
Universiti Putra Malaysia
43400 UPM Serdang
Selangor Darul Ehsan, Malaysia
shahida_jamail@yahoo.com, rodziah@fsktm.upm.edu.my

ABSTRACT

Nowadays, cloud computing is a very interesting environment for researcher to look deep. It is more on computing delivery whereby service rather than a product. It provides infrastructure as a service (IaaS), platform as a service (PaaS), software as a service (SaaS), and storage as a service. This technology is not involved end-user knowledge of the physical location and configuration of the system when users deliver the services. The cloud computing concepts is derived from grid computing, whereby end-user no need to understand the devices and components or infrastructure required in retrieved the service. It is based on virtualization resources. Cloud computing is consider as a new area in research and there have a lots of problem especially regarding to Service Level Agreement (SLA). This research is focusing to verify whether the SLA is follow the contract or violated. All the results are used for validation of the cloud services based on its own SLA.

KEYWORDS

Cloud computing, virtualization, Service Level Agreement (SLA), verification, validation, cloud services, contact.

1. INTRODUCTION

In cloud computing, Software Level Agreement (SLA) is one of the important parts which are dealing with services contractualization. [4].It can be saying that the cloud computing quite new and there a few researchers still looking with these issues. Because it is new, it raises many problems and most of them related to the implementation and deployment of Service Oriented Architecture (SOA) and virtualization at both hardware and software levels[4]. Besides that, cloud computing also raises questions related to the services contracts which are deal with Software Level Agreement (SLA).

SLA is a starting point to set the parameter and minimum levels for each elements of the service provider that will be required to meet the needed. These are also a very important since there are a lot of relationships between an organization and a cloud computing provider that will be contractually governed [3]. There are also important between consumers with cloud provider. All the contact is based on the negotiation between organization and service provider and consumers with cloud provider. The issues in a cloud computing that related to SLA are about service availability, response time performance, and

low security. In cloud computing there are always provide a very limited way of measuring the SLA parameters [1][2]. Usually this SLA will appear every time user uses the cloud services. But usually user or organization did not read the SLA thoroughly. Due to that, service provider can state any good specification but in reality the service did not perform as good as they mention in the SLA. The service provider unable to verify and validate the correct services or functions that requested from client in cloud computing by following the SLA. The cloud services always dealing with the violation of the SLA since there have no tools that can verify and validate that SLA.

Since there have lots of problem happen from the incorrect SLA, the tool to verify and validate the SLA will develop. The tool will verify whether the SLA is violated or not. In verification phase, the actual attributed that had been filter from the data collector where dealing with real time data will be compare with the attributes value in SLA. If some of the attributes value in the SLA is violated then the whole SLA is invalid and not follows the contract. If the attributes in the SLA is not violated then the SLA is valid and follow the contact. The results of the validation are important to notify the provider, service administrator and client whether the SLA is valid or invalid. The provider or service administrator will make a decisions regarding reconfiguration or terminate the contract and for client side, the result just for notification of the services.

This paper is organized as follows. Section 2 will describe about background of cloud computing and SLA. Section 3 presents the implementation of the verification of the SLA. Section 4 is about the experimental setup to verify the SLA attributes. Section 5 is to validate the SLA. Section 6 related works and finally section 7 for conclusion.

2. BACKGROUND

2.1 CLOUD COMPUTING

There are lots of researchers looking in the cloud computing issues. One of the most commonly used definitions of Cloud Computing is that of L. M. Vaquero [7]. It defines Cloud Computing as follows:

"Clouds are a large pool of easily usable and accessible virtualized resources (such as hardware, platforms and/or services). These resources can be dynamically reconfigured to adjust to a variable load (scale), allowing also for an optimum resource utilization. This pool of resources is typically exploited by a pay-per-use model in which guarantees are offered by the Infrastructure Provider by means of customized SLAs."

This definition clearly mentions for services checking in the context of Cloud Computing. Indeed, a Cloud provider must follow to this definition to make sure and guarantee the service levels for the provided resources must follow the contracts [5][6]. The contact is related to Service Level Agreement.

L. Alexander et al (2009) had proposed a generic Cloud computing stack that classifies Cloud technologies and services into different layers. They explain each layers clearly and demonstrate how the model helps in explaining the overall Cloud computing landscape and illustrates the use of the stack in modelling a Cloud computing ecosystem of various providers.[7]

2.2 SERVICE LEVEL AGREEMENT

A service-level agreement (SLA) is a part of a service contract where a service is formally defined based on the negotiation between provider and organization or provider and client. In other words, the term *SLA* is referring to the contracted the service and the performance. For example is Amazon.com. It is commonly include the Service Level Agreement as a contract between client and service provider.[3]

A Service Level Agreement is a document that stated the description of the service level parameter, service level objective, agreed service, guarantees and action in case of violation. A service-level agreement is a negotiated agreement that have documented between two parties which are customer and service provider. The Sla is very important to determine the availability, reliably, scalability of the services.

3. IMPLEMENTATION OF SLA VERIFICATION

This section describes the research methodology that includes the activities taken in the research. The framework of the research methodology is shown in figure 1.

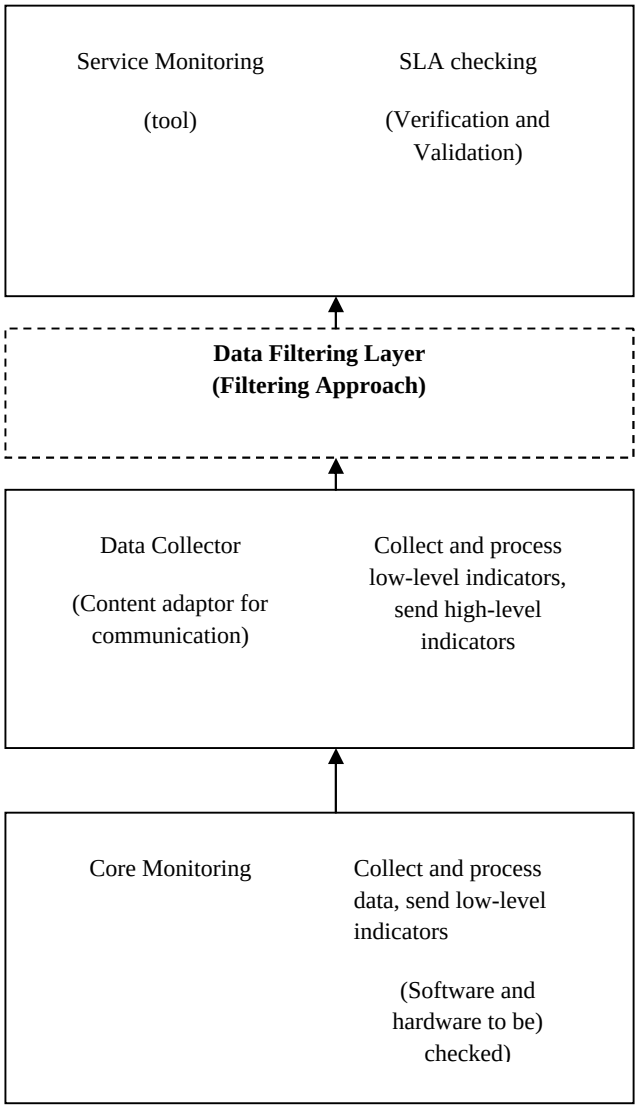

Figure 1. Framework of the proposed architecture

The propose framework is a three layer architecture and one Data Filtering Layer (Figure 1). This architecture included the Core Monitoring layer, the Data Collector layer and the Service Monitoring layer[1]. As shown in Figure 1, the lowest layer in this architecture is the Core Monitoring layer. This Core Monitoring will deal with infrastructure as a service (IaaS). All the hardware and software in core monitoring layer will interact with each other's and this layer will collect data from the various services and send to the low

level indicators. Such low-level indicators represent a view of the target service. Processing can take the form of data aggregation, transformation, enrichment, degradation, correlation, synchronization, etc. The Core Monitoring layer thus contains local or distributed.

The indicator values that produced before will transfer from the Core Monitoring to the Data Collector layers. The Data Collector layer will interact with both the Core Monitoring and the Service Monitoring layers but before entering Service Monitoring Layer, all the value will filter at Data Filtering Layer. The Data Collector is a tools where taking the input which is the low level indicators that was produced by the Core Monitoring layer[2]. These indicators are then processed in order to obtain values corresponding to higher-level business indicators. Such high-level indicators define a set of Service Level Objectives which can be specified in the SLAs related to the target service. All the data from Data Collector tools will be sent to Data filtering Layer whereby only certain Service Level Objective and SLA attributes will be used after filtering process. This process will apply the filtering approach. Only the need data will used and entered the Service Monitoring to processed. All the needed data based on the SLA attributes that most frequently stated in many SLA.

The upper layer is Service Monitoring layer. This layer is the highest level of the architecture. In this research, we only focus on Service Monitoring layer and a small part in Data Collector layer. The Service Monitoring layer will get the indicators value from Data Collector layer after all the data have been filtered by Data Filtering layer. The Service Monitoring layer contains a set of applications using monitoring data with various business concerns. Such applications include management applications, autonomous managers, and dynamic selection of services or performance qualification, verification and validation process of the SLA. They take as input high level indicators produced by the Data Collector layer after passing to the Data Filtering Layer. The Service Monitoring layer contains a tool whereby involved the verification and validation process for SLA.

The propose architecture, combined with the filtering approach known as Data Filtering Layer, offers a model for dealing with data filtering and verification and validation tools. Thus, a given SLA must at least define its identifier, matrics, and a non-empty set of Service Level Objectives (SLOs). A SLO must itself define at least its identifier (local to the SLA to which it belongs), the targeted information, a comparison operator, a value/information threshold and a description.

3.2 ARCHITECTURE OF VERIFICATION AND VALIDATION TOOL

The proposed architecture show that the collected data at the Data Collector layer will send to Service Monitoring layer. In this layer, there is several processes included Data Filter and SLA Checking. Below is the architecture of the tool.

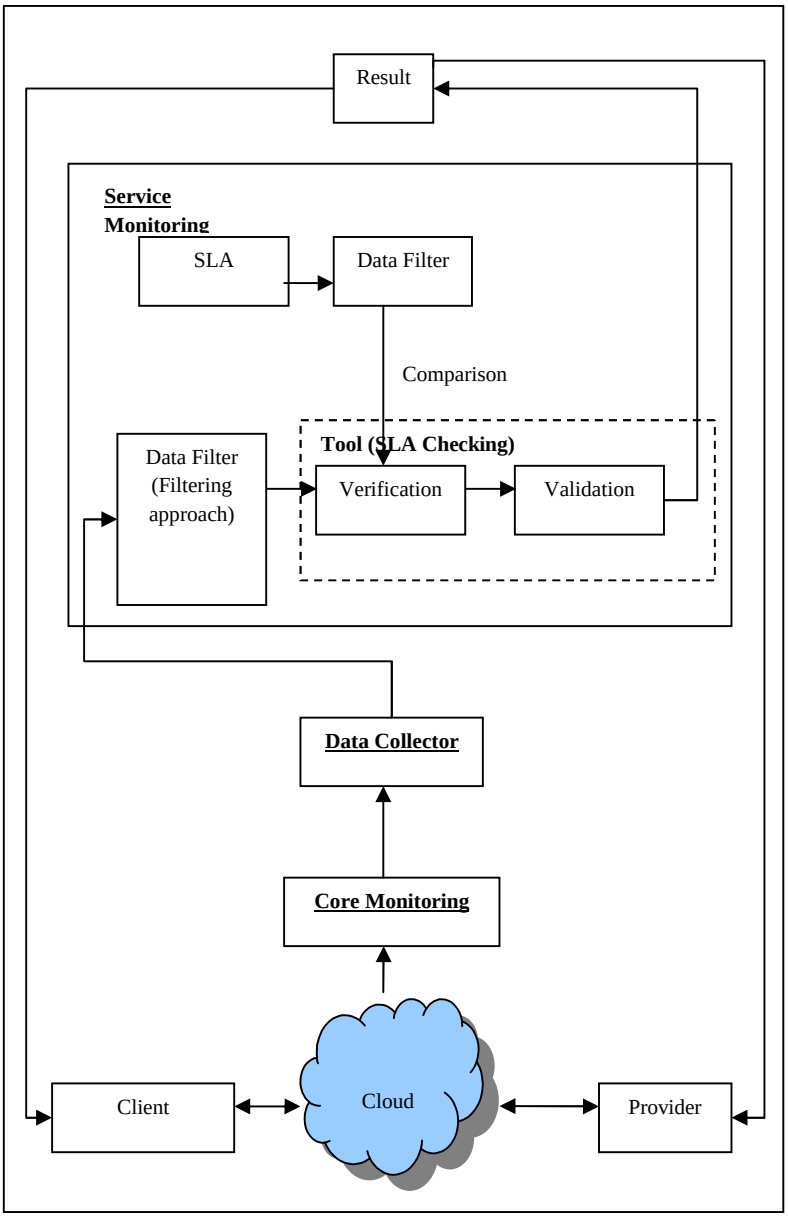

Figure 2. Architecture of Verification and Validation of the SLA (SLA Checking)

Once the data enter Service Monitoring layer, the data (real time data) will be filter by the Data Filtering layer. The Data Filtering layer will filter the data based on selected SLA attributes. Only the wanted data will get pass through this process. Then the output from the Data Filtering layer will be send to Service Monitoring for SLA checking.

Once the data enter the SLA checking tools, the data verification process will commence. The verification process is a process where the data received from Data Filtering layer (real time data) are being compared with the data that stated in the Service Level Agreement by using the comparison technique. If the received data is match with the SLA value then the result will show "In Specification". But if the data is not match then the result will show "Violation". Once the verification process complete, the result of the verification process will be supply to the next process which is Validation.

In this Validation process, the system will declare whether the software and/or hardware service is "Valid" or "Invalid". For that, the Validation process will scan the verification result. It will check if there is any violation result. If one Violation detected by the process, the service will be declared the software and/or hardware service as "Invalid", where else if no Violation result detected it will be declare as "Valid". Finally the validation process will produce a complete report which consists of software and/or hardware service actual performance status, verification result and the validation result. The validation results can indicate the SLA compliance, the violation and failure during the SLA verification or during the collection of monitoring information. For validation results in tool provided can be used to inform the target service administrator to make decisions regarding reconfiguration of the service or terminate a contract. Client also will get the notification of about the services whether the services follow the SLA or not. This main target of this research is to analyst whether the cloud services provided is following the contract or not. It

is very important in terms of reliability, security and availability.

4. EXPERIMENTAL SETUP FOR VERIFICATION OF SLA ATTRIBUTES

There are 2 ways in determine SLA attributes which are done by dry running and the other one is by survey approach. In this paper, we only justify for dry running results only. Figure 3 shown the flow to determine the SLA attributes through the dry running. First, 5 sets of SLAs from Amazon EC2, Googrid, Google, windowAzure and Salesforce.com will be reviewed. Based on the reviewed, the most frequent attributes that stated in all SLAs will be selected. The selected attributes will be analyzed and only attributed that have value will be chosen to compare with the results from survey approach.

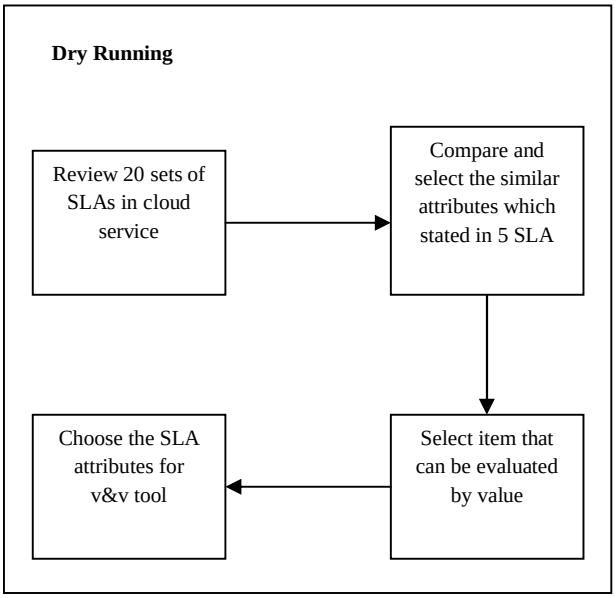

Figure 3. The Flow to Determine SLA Attribute through Dry Running

The second way to determine SLA attributes is through survey approach. Figure 4 is described the flow to determine SLA attributes through survey approach. The meeting or interview session with cloud expertise will setup. The discussion for selecting the SLA attributes is based on the experiences and user's preferences. All the attributes will be collected and analyze. Only the best attributes will be chosen. The results from the dry running will compare and analyze with the results from survey approach. The best attributes will be chosen for verification and validation process in tool.

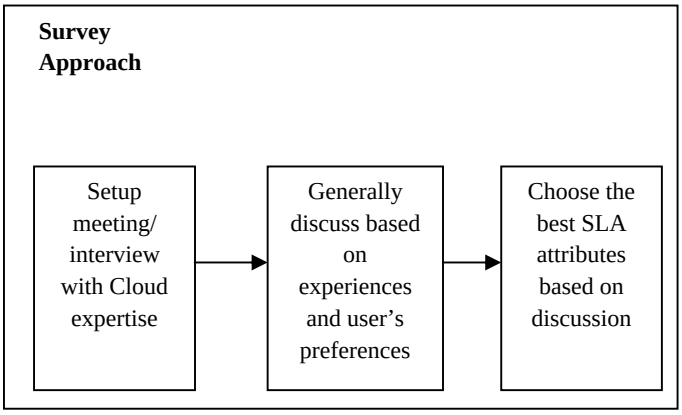

Figure 4. The Flow to Determine SLA Attribute through Survey Approach

5. RESULTS FROM DRY RUNNING PROCESS (PERFORMANCE METRICS)

The SLA parameters are determined based on proposed framework from A.Mohammed et. al (2010). 5 selected SLA which are from Amazon EC2, Salesforce.com, googrid, google and windowAzure are selected. All the SLA metrics / attributes will list [3]. Only the most friequently stated in SLA will be selected for SLA checking tool (verification and validation tool) There have four

categories in determinate the SLA metrics/ attributes which are in IaaS, PaaS, SaaS and storage as a service [3]. Below is all the results of dry running experiment.

Table 1. SLA attributes for IasS

Cloud Services / **Parameter Description**		**Gogrid**	**Google**	**Amazon EC2**	**WindowsAzure**	**Salesforce.com**
CPU capacity	CPU speed for VM			/		
Memory size	size Cash memory size for VM	/		/		
Boot time	time for MV to be ready for use					
Storage	Storage size of data for short or long term of contract	/		/		
Scale up	Maximum of VMs for one user	/		/	/	/
Scale down	Minimum number of VMs for one user					/
Scale up time	Time to increase a specific number of VMs	/	/	/	/	/
Scale down time	Time to decrease a specific number of VMs		/			/
Auto scaling	Boolean value for auto scaling feature			/		
Max number can be configured on physical server	Maximum number of VMs that can be run on individual server					
Availability	Uptime of service in specific time	/	/	/	/	/
Response time	Time to complete and receive the process	/		/	/	
Load Balancing	When elasticity kicks in	/		/		

Table 2. SLA attributes for PaaS

Cloud Services / **Parameter Description**		**Gogrid**	**Google**	**Amazon EC2**	**WindowsAzure**	**Salesforce.com**
Integration	Integration with eservices and other platforms	/				
Scalability	Degree of use with large number of online users	/		/		
Pay as you go billing	Charging based on resources or time of service	/	/	/	/	
Environments of deployment Servers	Supporting offline and cloud systems	/				
Browsers	Firefox, IExplorer,..	/		/		
Number of developers	How many developers can access to the platform					
Service Credit	Charging based on any failure	/	/	/		

Table 3. SLA attributes for SaaS

Cloud Services / Parameter Description		Google grid	Google	Amazon EC2	WindowsAzure	Salesforce.com
Reliability	Ability to keep operating in most cases	/		/	/	
Usability	Easy built-in user interfaces					
Scalability	Using with individual or large organisations	/		/		
Availability	Uptime of software for users in specific time	/	/	/	/	/
Customizability	Flexible to use with different types of users					

Table 4. SLA attributes for Storage as a service

Cloud Services / Parameter Description		Google grid	Google	Amazon EC2	WindowsAzure	Salesforce.com
Geographic location	Availability zones in which data are stored	/			/	
Scalability	Ability to increase or decrease storage space			/		
Storage space	Number of units of data storage	/	/			
Storage services (uptime)	Service of data storage	/	/	/	/	/
Storage billing	How the cost of storage is calculated	/		/	/	
Security	Cryptography for storage and transferring of data, authentication, and authorization	/		/		
Privacy	How the data will be stored and transferred	/				
Backup	How and where images of data are stored					
Recovery	Ability to recover data in disasters or failures					
System throughput	System response speed	/	/	/	/	/
Transferring bandwidth	The capacity of communication channels					
Data life cycle management	Managing data in data centres, and use of network infrastructure	/				

Most frequently applied in SLA

Frequently applied in SLA

Average applied in SLA

In this experiment, we determined that the most frequent attributes that applied in SLA are system throughput, storage service (uptime), availability and scale up time. System throughput is about the response time when client make request to cloud services. Storage service (uptime) is about the services of the data storage. For availability is more on uptime of the services for users in specific time and finally is scale up time. Scale up time is about the time to increase or decrease for the services based on how large number of users. All four Sla attributes will filter at Data Filtering Layer before entering the Service Monitoring layer.There have a lots of data that collected in Data Collector layer but only this four attributes will be filter and will used for SLA checking.

6. EXPECTED OUTCOMES

In this research, the Data Filtering Layer will apply to the SLA checking tools. The layer will filter all the data from the Data Collector layer. Only the important and needed data will used and the unknown data will ignore. For the existing research, there

did not filter the data from the Data Collector Layer. All the need and unneeded data will be process in Service Monitoring layer. This will cause the higher response time compare then by applying the Data Filtering layer. Only the need data from the Data Collector layer will processes at Service Monitoring for SLA checking. Below table is the example of results by applying the Data Filtering layer to verify and validate the SLA in cloud services.

Table 5. The Expected Result by Applying The Data Filtering Layer

Service No.	Time Taken with Data Filtering (ms)	Time without Data Filtering (ms)
1	5	15
2	9	20
3	10	22
4	12	40
5	15	45
6	20	49
7	25	50
8	26	55
9	35	62
10	43	63

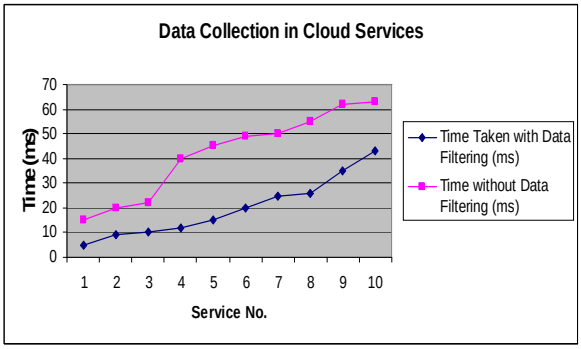

Figure 5. The Expected Result by Applying The Data Filtering Layer

Based on the graph above, it's clearly shown that by applying the Data Filtering layer can cause the lower response time in collecting the data. Compare to the existing research that cause the higher response time in collecting the data. This is because it will collect all the needed and unneeded data and pass to the verification and validation tool (SLA checking). By applying the Data Filtering layer it can prove that it will take the lower response time because only the needed data will pass for SLA checking for verify and validate the real time data with the SLA data.

Below table shown the expected result for verify and validate the SLA for cloud services. The verify phase will use the comparison technique where the actual value that collected from the Data Filtering Layer will compare to the SLA value from the cloud services. If the actual value smaller than SLA value then the SLA might be violate. If the actual value is same and bigger than SLA value then the SLA is correct and follows the SLA rules in cloud services.

The violate SLA is consider invalid in validation phase The SLA is valid if the SLA is correct and follow the rules based on the actual value.

Table 6. The Expected Results in Verify and Validate the 10 SLA in Different Cloud Services in terms of Security.

SLA No.	Actual value (%)	SLA value (%)	Comparison Result (verify)	Evaluate (validate)
1	45	100	Violate	Invalid
2	100	95	Not violate	Valid
3	80	100	Violate	Invalid
4	100	100	Not violate	Valid
5	40	100	Violate	Invalid
6	70	100	Violate	Invalid
7	100	80	Not violate	Valid
8	48	100	Violate	Invalid
9	100	100	Not violate	Valid
10	59	100	Violate	Invalid

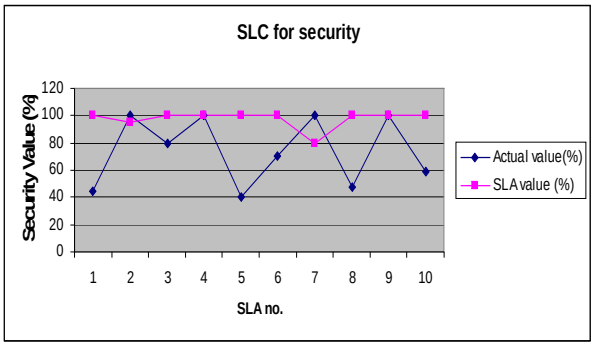

Figure 6. The Expected Results in Verify and Validate the 10 SLA in Different Cloud Services in terms of security.

Based on the graph above, it is clearly shown that if the actual value (real time data) lower than the SLA value it is mean that the SLO is violated. There have 6 violated SLO and four with not violated. When the SLO is violated automatically the SLA is invalid and not follows the contract. If the actual value is more or similar to SLA value means that the SLO is not violated and the SLA is valid and follow the contract.

7. RELATED WORK

The analysis of the SLA attributes is based on the existing solution from A.Mohammad et. al (2010). They design the framework of the SLA parameters for cloud computing concepts. There have four parts in determining the SLA parameter where are IaaS, PaaS, Saas and storage as a service. From four parts of the SLA, the most parameters that consumers can use to create a reliable model of negotiation with the service provider will determine.[3]

8. CONCLUSION AND FUTURE WORK

In this research, the tool for verification and validation of SLA in cloud services will develop. There have four layers that will create which are Core Monitoring, Data Collector, data filtering and service monitoring (verification and validation tool) but in this research we only focus on the Service Monitoring layer and Data Filtering layer [2]. All the data collected in the Data Collector will be filter at Data Filtering layer. Only the needed data will be used to verify and validate with the SLA in cloud services. The verification and validation tool should be able to verify the SLA is violated or not. If the SLA is violated then the results should be invalid and the SLA is not following the contract. If the SLA is not violated then the results is valid and the SLA is correct and follow the contract. The validation results are used to inform the target service administrator to make decisions regarding reconfiguration or to terminate a contract. The results also will goes to client for notification purpose. In future works we will deploy the real SLA checking tools to deal with real data and real cloud services. We are targeting the cloud

environment in Universiti Putra Malaysia for experimental setup.

9. ACKNOWLEDGEMENTS

We would like to thank to Associate Professor Dr Rodziah Atan for her help to guide and throughout this work.

10. REFERENCES

1. Chazalet, A : Service Level Checking in the Cloud Computing Context, Software Engineering Advances, pp. 297-304, IEEE 2010.
2. Chazalet, A : Service Level Agreement Compliance Checking in the Cloud Computing: Architectural Pattern, Prototype, and Validation. Software Engineering Advances, pp. 184-189, IEEE 2010.
3. Alhamad, M , Dillon, T, Chang, E : Conceptual SLA Framework for Cloud Computing, pp, 606-610, IEEE 2010.
4. Armbrust, M, Fox, A, Griffith, R, Joseph, A, Katz, R, Konwinski, A, Lee, G, Patterson, D, Rabkin, A, Stoica, I, Zaharia, M , Above the clouds: A Berkeley view of cloud computing. EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2009-28, 2009.
5. Hilley, D., Cloud Computing: A Taxonomy of Platform and Infrastructure-level Offerings. 2009.
6. Rupach, P, Sobolewshi, M, Dynamic SLA Negotiation in Autonomic Federated Environments. 2009.
7. Vaquero, L, Merino, L, Caceras, J, Lindner, M, "A break in the clouds: towards a cloud definition", ACM SIGCOMM Computer Communication Review, vol. 39, issue 1, Jan. 2009.