CloudWeb: A Web-based Prototype for Simulation of Cross-Cloud Communication Framework (C3F)

Ahmad Waqas, M. Abdul Rehman, Abdul Rehman Gilal, Mohammad Asif Khan

***Abstract* – Cloud computing has emerged as a solution for large-scale online service applications that require intense processing, storage and networking capabilities. Cross-cloud communication framework (C3F) was proposed to enable intercommunication among clouds in order to share computing resources. For testing of C3F requires simulation of multiple clouds simultaneously. Although, many cloud simulators are available but multiple cloud simulation under different policies and administrations is not supported. This paper discusses the development of CloudWeb that is a web-based prototype for simulation of multiple clouds concurrently. CloudWeb is developed using open source technologies.**

***Index Term* - Cloud Computing, Cloud Simulation, Cloud Network, Cloud Communication, Web Services.**

I. INTRODUCTION

With the rapid development of Internet technologies, recent years have witnessed the rise of large-scale online service applications such as web search, social networking and content delivery. As these applications require significant networking, processing and storage capacities, it is a critical challenge to design large-scale computing infrastructures for supporting these applications in a cost-effective manner. As a solution, cloud computing emerged during the last decade and became an attractive business for the companies and organizations that own large datacentres to rent their computing resources[1][2][3]. It is a computing paradigm to host and deliver computing resources over the Internet that has evolved promptly and captured the current business market. As a result, multi-billion dollar organizations such as IBM[4], Amazon [5], Google[6], and last but not least, eBay have hugely capitalized in cloud technology to offer cloud services.

Efficient resource management and allocation, cloud security, and cloud architecture implementation are primary research interests among cloud researchers. During last decade, many research groups have been formed and lots of research has been published in this field of study. The research for cloud infrastructure and implementation, got special attention and the present cloud implementations follow NIST reference architecture[7]. Besides this, intercommunication among clouds did not get much consideration even though it can benefit in several ways. For instance, organizations that manage their private clouds and have common business interests, can connect to form a network of clouds. Some advantages of this intercommunication are sharing of resources and attacks information. For example, in certain situation, it happens that the requested resource is unavailable for allocation at a particular cloud and denying the client request may cause business loss[8]. In such situations, a cloud may request resources from its connected clouds from network for allocation to its clients.

This above discussed solution can only be applicable to the clouds that form a network. To the best of our knowledge, the present cloud implementations are secluded and do not allow the interconnection among different clouds that is a bottleneck to form the cloud network. To tackle this, a cross-cloud communication framework (C3F) is proposed that is discussed in proceeding sections. As the present cloud implementations do not connect, hence the present cloud simulators do not allow the simulation of multiple clouds that can enable the inter-cloud communication. This paper aims to discuss a working web-based prototype for facilitating the simulation of multiple clouds concurrently that also enables cross-cloud communication.

II. CROSS-CLOUD COMMUNICATION FRAMEWORK (C3F)

National Institute of Standards and Technology (NIST) presented a Cloud Computing Reference Architecture and Taxonomy [7] to provide a general framework for accurate communication of the components and offerings of cloud computing. It discusses the five main actors of cloud computing that includes cloud consumer, cloud broker, cloud service provider, cloud auditor and cloud carrier. Figure 1 portrays an overview of present implementations of cloud computing infrastructure following the NIST definition of cloud computing [9]. Although the NIST architecture is a generic framework to implement cloud services, it does not support the integration and communication among multiple clouds under different rules, policies and administration. Mostly, the implementations of clouds are isolated and lack for intercommunication among multiple clouds.

The cross-cloud communication framework (C3F) is an extension of the current cloud implementations based on the guidelines provided by the NIST cloud computing reference architecture [7]. The distinguishing feature of C3F is to enable the inter-cloud communication. The current cloud implementations are usually isolated from each other that causes lack in communication among different clouds. However, the inter-cloud communication can benefit in many ways but two are major: Sharing of resources and attack information. For instance, if a cloud is overloaded with client requests and the resources are over utilized then a cloud may borrow resources form other clouds which are running with underutilized resources based on some agreed SLA [10]. Similarly, if one cloud is attacked with some

Ahmad Waqas, M. Abdul Rehman, Abdul Rehman Gilal and Mohammad Asif Khan are with Department of Computer Science, IBA University, Sukkur, Pakistan.
Email: ahmad.waqas@iba-suk.edu.pk, rehman@iba-suk.edu.pk, a-rehman@iba-suk. .edu.pk, asif.khan@iba-suk.edu.pk.

Manuscript received Nov 29,2016; revised on revised Dec 13,2016 ; accepted on Dec 16, 2016.

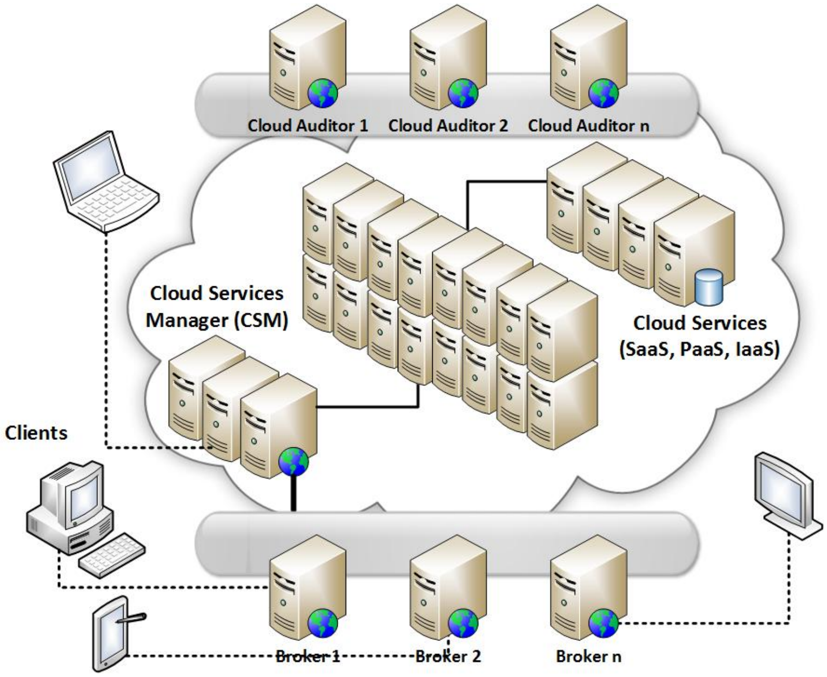

Fig. 1 Present Cloud Implementations

security breach to mutilation of the cloud resources, then there are likely chances that the other cloud may also confront the same attack[11]. Thus, the cloud can secure itself if it has the information of attacking entity well in-time [12]. For that, inter-cloud communication can help to share information of attacks among clouds to protect them from the same attacks with well in-time information.

To accomplish the intercommunication between clouds with different rules, policies and administration, C3F incorporate a component named "Inter-Cloud Communication Manager" in the cloud architecture as exhibited in Figure 2. The Cloud Service Manager (CSM) is one of the important fault tolerant, distributed architectural component and a single point of entry that plays the role of a bridge between client and the services. A client may request for the utilization of some service directly to CSM or via a cloud broker. The CSM is responsible to entertain all the requests from clients and brokers. The situation may arise that the requested resource is not available at that point in time, then CSM will coordinate with the Inter-Cloud Communication Manager (ICM) for borrowing the resource from connected clouds. Similarly, CSM is responsible to coordinate with ICM for lending the resources to the connected clouds. The Inter-Cloud Communication Manager (ICM) enables the connection between clouds having different rules, policies and administration. It facilitates the cross-cloud communication for benefiting with sharing of resources and attacks information. The cloud manager of different clouds can be connected together to form a network of clouds as illustrated in Figure 2. The ICM of all four clouds are connected and the communication between these clouds can be performed. The ICM is responsible to maintain the Key Table that contains the information of connected clouds. ICM is also responsible to coordinate with CSM to maintain the log file for all the external events concerned with cloud environment.

III. RELATED WORK

Cloud simulators play an important role and facilitate researchers for rapid evaluation of the efficiency, performance, and reliability of new algorithms on large heterogeneous cloud infrastructures. Some cloud simulators are commercial and some are open source, for example, CloudSim [13], CloudAnalyst [14], GreenCloud [15], iCanCloud [16], MDCSim [17], DCSim [18], EMUSIM [19] and D-Cloud [20]. However, these simulators emphasis on the simulation of specific cloud computing components. As an example, some targets the simulation of large-scale datacentres [13], other simulates the cloud applications and analyse their behaviour [14], and few focuses on the workload distribution, and fault tolerance analysis [15]. To the best of our knowledge, no simulator facilitates with simulation of multiple clouds simultaneously. For implementing and testing the above discussed C3F and experimentations, multiple clouds need to be available at the same time in order to perform communication among clouds. Such a simulation technique for multiple cloud simulation is presented in [21].

IV. WEB-BASED PROTOTYPE AND SIMULATION OF C3F

The main objectives of C3F are resource sharing and disseminating attacks' information among clouds. Developing a complete cloud on virtual machine is expensive in terms of higher cost and time. Therefore, a prototype is developed for testing and validation of C3F as

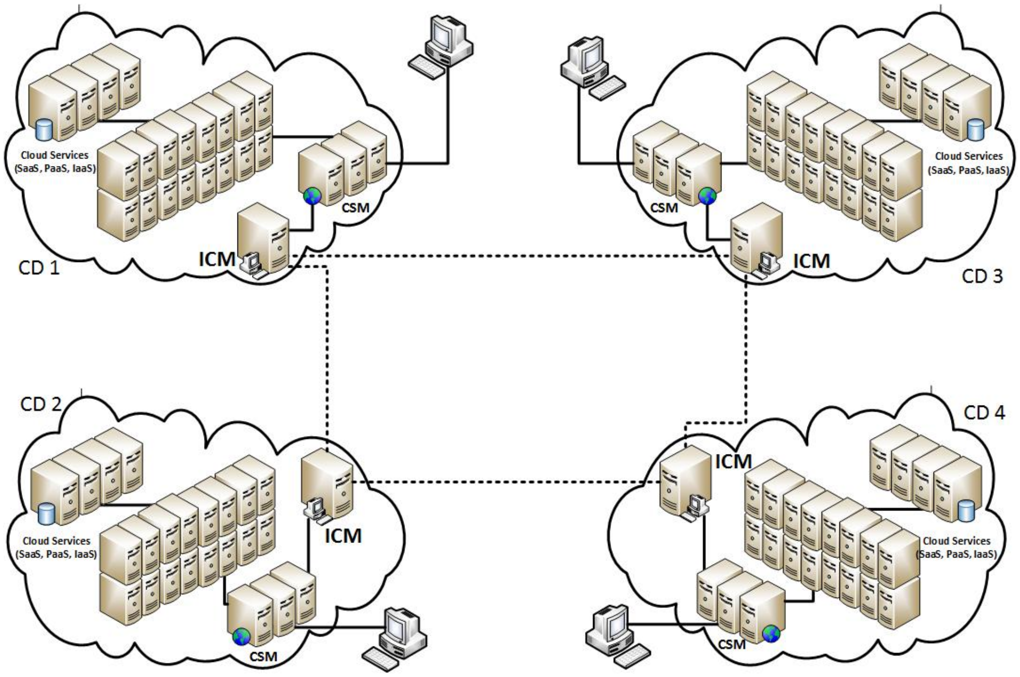

Fig. 2 Cross-Cloud Communication Framework [10]

practiced by other cloud researchers [22]. The development of web based prototype was logically divided into two phases: algorithmic design and implementation. In algorithmic design, the low-level description technique is used to define the algorithms of the prototype. Implementation of these algorithms are conducted by using open-source technologies.

For that purpose, Apache version 2.2.29 was used as a webserver, Hypertext Pre-processor (PHP) language as a server-scripting language, Java scripting language for client side scripting, and MySQL was used as relational database management system (RDBMS). Fig. 3 illustrates the phases of the prototype. In C3F, three basic actors are involved: Client, CSM, and ICM. Both, CSM and ICM, actors were setup on different servers. Four different clouds, each of them was with one CSM and ICM, were developed and setup separately. All of these clouds were programmed to allow inter-communication between each other to share resources and attack information as shown in Figure 4.

There are three main components of the prototype: interface, communication, and database. The following sections discuss the components in detail.

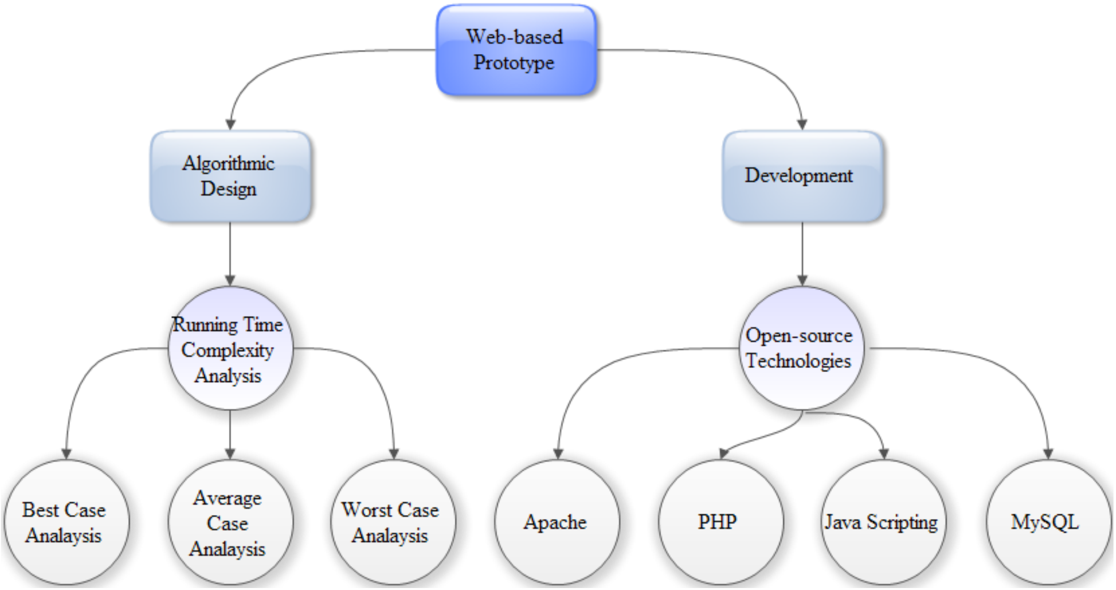

Fig. 3 Development Phases for Web-based Prototype

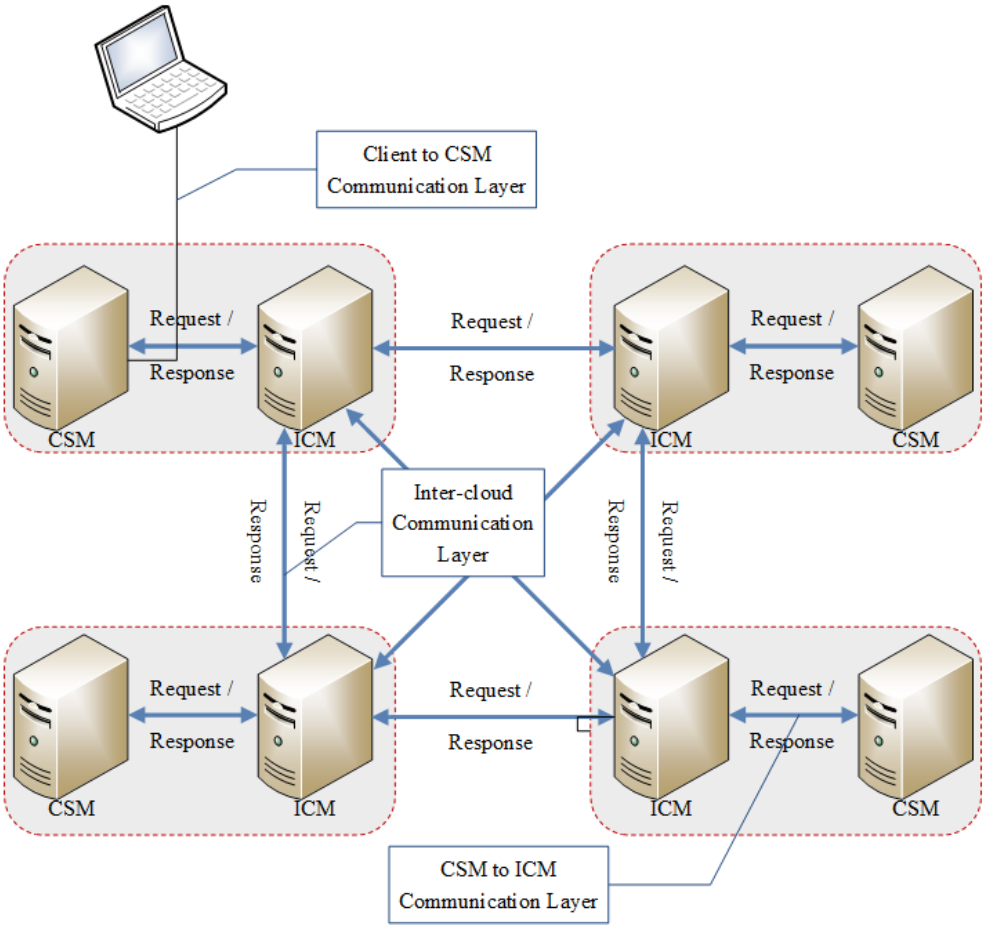
Client to CSM Communication Layer
Request /
Response
CSM
ICM
Inter-cloud Communication Layer
Response
Request /
CSM to ICM Communication Layer

Fig. 4 Experimental Cloud Setup

A. User Interface

The development of the prototype was divided based on the actors of the C3F. Technically, this prototype has provided interaction interface to client users only. The cloud manager and the cloud service manager contained only automated activities, therefore, no physical interface was designed for them.

The interface enabled the client to perform certain activities: login, signup, request and terminate services, and payment as illustrated in Figure 5. For testing purpose, in this prototype, only basic information was received from client for login: email and password. Moreover, resource request is totally dependent on the signup form in which client is asked to agree the SLA and select the resources for future use. Therefore, the signup activity was also divided into three sub-activities as described in Table 1.

Finally, the termination activity was considered to be developed to show the overall usages of the resources by client. Therefore, the termination activity was coded to calculate the overall charges to be paid, and history of the payments.

Table I. Signup activities

Activity	Description
Receive Biographic details	Name, Email, passwords, etc.
SLA	Acquire digital signatures from user for SLA terms and conditions.
Select Services	Choosing services for future use based on agreed prices and usage details.

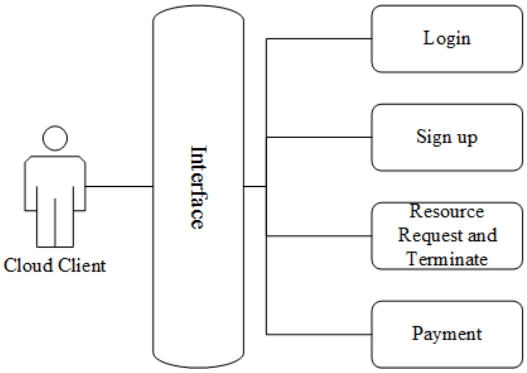

Fig. 5 Cloud Client User Interface

B. *Communication*

The term communication here refers to the systematic processes of sharing resources and attacks information. Basically, this prototype established two-way communication among Client to CSM, CSM to ICM, and ICM to ICM. It should be noted that client can never communicate with ICM directly. It should also be kept into account that all communications can never be done without login acceptance. Hence, the following sections discuss the communication activities without mentioning the login and signup activities.

C. *Client to CSM*

A client has to request to CSM in order to use resources from cloud. On the other hand, CSM is also supposed to either provide requested services or terminate the services based on the actions. Therefore, this section of development has implemented activities listed as in Table 2.

Resource request and response are always considered basic activities of the clouds. But, in C3F, “Block client” is one of the main activities of CSM. It generally triggers on two situations, either CSM finds a client guilty or CSM is notified by interconnected ICM.

D. *CSM to ICM*

A request from client always forwarded to ICM if the requested resource is currently not available at CSM. On the other situation, ICM can also communicate to its CSM if the ICM has received a resource request from connected clouds. It is because the communication is bi-directional. Therefore, CSM and ICM both can request and response each other. Basically, implementation of this layer was conducted by following the algorithms of resource sharing and attacks information dissemination. The main activities that were developed to be performed on this layer are discussed in Table 3.

The activities mentioned above were coded independently with involvement of servers (CSM and ICM) by using session methods (especially curl_setopt() in PHP) because each activity needs some responses for further actions. For instance, “Request to borrow resource” is generated at CSM to ICM, and client is kept on hold until the ICM responds to CSM after receiving resources from cloud network. By this way, each server maintains their own security levels and processes.

E. *ICM to ICM*

A layer of communication between ICM to ICM can only occur when, either, ICM has received request from its CSM to look for a resource from connected clouds (from key table) or during attack information sharing. This layer is basically a bridge to create interconnection between different clouds. Generally, the similar standards of development were followed as at CSM to ICM communication layer. In this study, ICM is also considered as an independent server same like CSM.

At this communication layer, both ICMs have got the same rights of communication as the communication layer establishment is solely based on the SLA. Therefore, defining one-way communication in this section can make sense of another way communication at the same time. For example, if ICM of cloud 1 shares attacks information to interconnected clouds and, it shows, ICM of cloud 2 can also share attacks information to it. Table 4 discusses the activities developed for ICM to ICM communication layer.

It should be noted here that for efficient network traffic management the “response for resource” activity is obliged only if the CSM indicates the availability.

Table II. Client to CSM Activities

Activity	**Actor**	**Description**
Request Resource	Client	Client can request any agreed resource from CSM.
Request Terminate Resource	Client	Client can request for service termination from CSM.
Payment	Client	Client can request or receive the payment invoice and payment history from CSM.
Response Resource	CSM	CSM can allocate the resource based on availability and SLA.
Terminate the Resource	CSM	CSM can make termination of use based on client request
Response Payment	CSM	CSM can make payment invoice
Block Client	CSM	CSM can block the client if found vulnerable

F. *Database*

It was important to record all activities performed by any of the actors, either by client or automated machines (CSM and ICM). In this study, different actors were implemented on different servers. For example, all four ICMs were implemented and setup on different machines. Meanwhile, activities related to Clients and CSM, and for one cloud were coded on one machine. For this prototype, one cloud possesses two databases, independently, in which one is for Client and CSM and another for ICM.

Database for client and CSM was created to store data for three main purposes: Client information, CSM activities, and record history. The client information space was occupied to store basic information of client such as names, emails, contact numbers, and payment details. Whereas, CSM activity space was kept to store information related with client’s SLA, resource details (i.e., availability or engaged by which client), payment management, and vulnerable or blocking information. Lastly, each activity performed, by any actor on server was carefully stored into log file to monitor the history.

ICM database was also created with three logical pairs: CSM activities, ICM activities, and record history. Basically, CSM activity space was divided to store details of communication between ICM and CSM. For this purpose, resource request and response details can be stored in that space. In the same way, ICM activity space was separated to store information regarding interconnected clouds and SLAs between clouds. For instance, interconnected clouds details are stored in Key Table to manage inter-cloud communication. Finally, all activities between CSM to ICM and ICM to ICM were monitored into log files.

Table III. CSM to ICM Activities

Activity	Actor	Description
Request to borrow resource	CSM	CSM can request to its ICM if the resource is currently not available to allocate to a client
Terminate borrow Resource	CSM	CSM can terminate the borrowed request at any time requested by client.
Calculate the Payment	CSM	CSM is responsible to calculate the summary and history of payment at its end based on SLA.
Forward vulnerability information	CSM	CSM immediately forwards the vulnerable activity performed by any entity to its ICM to disseminate to connected clouds.
Response resource availability	CSM	If CSM is requested by ICM to provide a resource for connected cloud then CSM can allocate or deny based on availability.
Response borrow resource request	ICM	ICM responds to CSM upon receiving or rejection of requested resource from interconnected clouds.
Request to lend a resource	ICM	ICM can forward lend request to its CSM in order to facilitate resource to borrower cloud (based on availability)
Vulnerable entity Information	ICM	On receiving, ICM immediately notifies the vulnerable entity information to CSM for further actions.

V. RESULTS AND DISCUSSION

The C3F was simulated using the CloudWeb considering the simulation conditions discussed in previous section. Four clouds were designed and networked for the sharing of resources and attacks information. These clouds were enabled with ICM to facilitate the cross-cloud communication and were closely observed.

For the resource sharing processes, the performance of CloudWeb was observed based on success rate and allocation time. The success rate generally refers to the successful allocation of requested resource when it is available either from local cloud or foreign cloud, but more specifically, after borrowing the resource from cloud network. Whereas, Resource allocation time measures the time taken to deliver a resource after the request is received. The resource allocation time is computed in microseconds and the total performance cost T_c of requested resources allocation is computed by the following equation.

$$T_c = \sum_{i=1}^{n}(\tau \times \psi) + \sum_{j=1}^{m}(\tau \times \phi)$$

Where,

$n =$ The number of requests with successful allocation
$m =$ The number of requests with unsuccessful allocation
$\tau =$ The total time taken for processing request
$\psi =$ The success rate
$\phi =$ The unsuccessful rate

The observations showed that 94.4% of the time, client's request was successfully fulfilled by borrowing the resource from other clouds that is because of successful cross-cloud communication among cloud network. The mean allocation time was calculated 12 microseconds when resource is allocated from local cloud. An increase in allocation time is noticed when it is borrowed from other connected clouds. Whereas, the mean allocation time for borrowing and allocating resource from all connected clouds showed a minor difference.

An important concept of C3F is to disseminate the attacks information whenever a cloud detects the attacks so that the other clouds may protect themselves from the same attack and intruding entity. The CloudWeb is used to further simulate for the dissemination of attack's information. For this, random attacks were created on different clouds to observe the results and 100% success rate was observed for sharing the attacks' information and blocking the intruding entities on all connected clouds. These attacks were created randomly on different clouds simultaneously and with controlled time span too.

VI. CONCLUSION

Cloud simulators are important to test and validate processes and algorithms. Intercommunication among cloud network is also important in order to share computing resources for load balancing and fulfilling the requests of client. Moreover, it benefits with sharing of attacks information so that other clouds may protect themselves from similar attacks. In this paper, we presented a prototype for testing of inter cloud communication by programming cloud manager and inter-cloud communication manager as separate servers on different machines. CloudWeb facilitates the simulation of simultaneous multiple clouds.

Table IV. ICM to ICM Activities

Activity	Actor	Description
Request for resource	ICM	ICM can broadcast resource request to connected clouds (based on key table)
Response for Resource	ICM	ICM can make response to requested request after checking availability with its CSM.
Broadcast vulnerable information	ICM	ICM has responsibility to broadcast the vulnerable details to connected clouds as immediately after receiving by CSM.
Receive vulnerability information	ICM	ICM can also be notified by connected clouds for vulnerable information which can be used by CSM to avoid threats.

REFERENCES

[1] R. Buyya, C. S. Yeo, and S. Venugopal, "Market-oriented cloud computing: Vision, hype, and reality for delivering IT services as computing utilities," in 10th IEEE International Conference on High Performance Computing and Communications, HPCC 2008, 2008, pp. 5–13.

[2] B. Prasad, R. Admela, D. Katsaros, and Y. Goeleven, "Architectural Requirements for Cloud Computing Systems : An Enterprise Cloud Approach," J. Grid Comput. Springer, vol. 9, no. 3, pp. 3–25, 2010.

[3] W.-T. Tsai, X. Sun, and J. Balasooriya, "Service-Oriented Cloud Computing Architecture," in Seventh International Conference on Information Technology, 2010, pp. 684–689.

[4] M. IBM Cloud, "Tap into expertise and bring ideas to life with hundreds of IBM and partner services," 2008. [Online]. Available: http://www.ibm.com/marketplace/cloud/us/en-us. [Accessed: 30-Apr-2015].

[5] Amazon, "AWS Cloud Computing,". [Online]. Available: https://aws.amazon.com. [Accessed: 30-Apr-2017].

[6] GoogleCloud, "Google Cloud Platform," [Online]. Available: https://cloud.google.com/. [Accessed: 30-Apr-2017].

[7] F. Liu, J. Tong, J. Mao, R. Bohn, J. Messina, L. Badger, and D. Leaf, "NIST Cloud Computing Reference Architecture: Recommendations of the National Institute of Standards and Technology," in NIST Special Publication 500-292, Cloud Computing Program Information Technology Laboratory National Institute of Standards and Technology Gaithersburg, MD 20899-8930, 2011, pp. 1–35.

[8] A. Nathani, S. Chaudhary, and G. Somani, "Policy based resource allocation in IaaS cloud," Futur. Gener. Comput. Syst., vol. 28, no. 1, pp. 94–103, 2012.

[9] P. Mell and T. Grance, "The NIST Definition of Cloud Computing (Draft) Recommendations of the National Institute of Standards and Technology," in NIST Special Publication 800-145 (Draft), Computer Security Division, Information Technology Laboratory (ITL), National Institute of Standards and Technology (NIST), U.S. Department of Commerce, Gaithersburg, MD, USA., 2011.

[10] A. Waqas, Z. M. Yusof, A. Shah, and M. A. Khan, "ReSA : Architecture for Resources Sharing Between Clouds," in Conference on Information Assurance and Cyber Security (CIACS2014), 2014, pp. 23–28.

[11] A. Waqas, Z. M. Yusof, and A. Shah, "A security-based survey and classification of Cloud Architectures, State of Art and Future Directions," in 2nd International Conference on Advanced Computer Science Applications and Technologies – ACSAT2013, 2013, pp. 284–289.

[12] A. Waqas, Z. M. Yusof, A. Shah, and N. Mahmood, "Sharing of Attacks Information across Clouds for Improving Security : A Conceptual Framework," in IEEE 2014 International Conference on Computer, Communication, and Control Technology (I4CT 2014), 2014, pp. 255–260.

[13] R. N. Calheiros, R. Ranjan, A. Beloglazov, and A. F. De Rose, "CloudSim : a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," Softw. – Pract. Exp., vol. 41, no. 1, pp. 23–50, 2011.

[14] B. Wickremasinghe, R. N. Calheiros, and R. Buyya, "CloudAnalyst: A CloudSim-Based Visual Modeller for Analysing Cloud Computing Environments and Applications," in 24th IEEE International Conference on Advanced Information Networking and Applications, 2010, pp. 446–452.

[15] D. Kliazovich, P. Bouvry, and S. U. Khan, "GreenCloud: a packet-level simulator of energy-aware cloud computing data centers," J. Supercomput., vol. 62, no. 3, pp. 1263–1283, Nov. 2010.

[16] A. Núñez, J. L. Vázquez-Poletti, A. C. Caminero, G. G. Castañé, J. Carretero, and I. M. Llorente, "iCanCloud: A Flexible and Scalable Cloud Infrastructure Simulator," J. Grid Comput., vol. 10, no. 1, pp. 185–209, Apr. 2012.

[17] S.-H. Lim, B. Sharma, G. Nam, E. K. Kim, and C. R. Das, "MDCSim: A multi-tier data center simulation, platform," in 2009 IEEE International Conference on Cluster Computing and Workshops, 2009, pp. 1–9.

[18] M. Tighe, G. Keller, M. Bauer, and H. Lutfiyya, "DCSim : A Data Centre Simulation Tool for Evaluating Dynamic Virtualized Resource Management," in 2012 8th international conference on Network and service management (cnsm), and 2012 workshop on systems virtualiztion management (svm), 2012.

[19] R. N. Calheiros, M. A. S. Netto, and R. Buyya, "EMUSIM : An Integrated Emulation and Simulation Environment for Modeling , Evaluation , and Validation of Performance of Cloud Computing Applications," Softw. – Pract. Exp., vol. 0, no. 1, pp. 1–18, 2012.

[20] T. Banzai, H. Koizumi, R. Kanbayashi, T. Imada, T. Hanawa, and M. Sato, "D-Cloud : Design of a Software Testing Environment for Reliable Distributed Systems Using Cloud Computing Technology," in 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing, 2010, pp. 631–636.

[21] A. Waqas, Z. M. Yusof, A. Shah, Z. Bhatti, and N. Mahmood, "Simulation of Resource Sharing Architecture between Clouds (ReSA) using Java Programming," in 2014 International Conference on Information and Communication Technology for Muslim World (ICT4M), 2014, pp. 5–10.

[22] A. García-García, I. Blanquer Espert, and V. Hernández García, "SLA-driven dynamic cloud resource management," Futur. Gener. Comput. Syst., vol. 31, no. 1, pp. 1–11, 2014