BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **14**, No 3

Sofia • 2014

Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.2478/cait-2014-0034

A Virtual Firewall Mechanism Using Army Nodes to Protect Cloud Infrastructure from DDoS Attacks

N. Jeyanthi, P. C. Mogankumar

School of Information Technology and Engineering,
VIT University, Vellore-632014, Tamilnadu, India
E-mail: njeyanthi@vit.ac.in

Abstract: *Cloud is not exempted from the vulnerability of Distributed Denial of Service (DDoS) attack, a serious threat to any distributed network and has considerably less effective solutions to deploy in the network. This paper introduces a novel mechanism to protect and prevent the cloud from the spurious packets targeting the depletion of server resources. The army nodes called "Cloud DDoS Attack Protection" (CDAP) nodes are installed at the cloud server farm/ Datacenter (DC). These army nodes act as virtual firewall without destroying the Cloud Infrastructure and improve the availability of DC, even at the time of DDoS attack. By continuously monitoring the incoming packets, CDAP filters the attack packets intruding the Cloud DC. Availability is further improved by handing over the threat detection and attack mitigation to CDAP nodes and by redirecting the malicious user requests to the dump network. The simulation results prove that the introduction of CDAP nodes improve the availability and reduce the response time and the cost incurred.*

Keywords: *DDoS, CDAP, Datacenter, Virtual Firewall, Cloud computing.*

1. Introduction

Resource sharing, scalability, multi-tenancy and virtualization characteristics of Cloud computing relieved the industries from huge investments. As more and more users are attracted towards the cloud, security issues pose challenges to the users as well as providers.

Recent surveys reveal that DDoS is the most hazardous security threat to all types of networks and Cloud is not exempted. The fast flux technique at DNS and

its organization helps in improving the load balancing policy. DataCenters are the resource provisioners where the usage of each and every resource, such as Bandwidth, Random Access Memory, Virtual Machine, Storage, data processing servers and other resources, is for profit. If these DataCenters are attacked, the load balancing policy helps in downgrading the attack but is not defending against the attack. This leads to huge bandwidth loss and also fails to serve legitimate users. DataCenters are the shared resource pool and hence they are always in demand. If the DataCenter suffers from DDoS attack, then the requested service cannot be provided. Eventually this leads to loss of fame, customer goodwill and profit (in terms of loss of resources and cost).

Although Cloud computing technology can defend against this attack, the exponential hike in the rate of attack makes it difficult to withstand. DDoS attacks can be initiated by a group of distributed human attackers or botnets. In this paper, a new proposed scheme named CDAP, also called as Army nodes, is presented. Army nodes are deployed in order to prevent the attack packets from entering the cloud network.

The proposed work is structured as follows: Section 2 presents related work. Section 3 introduces the architecture overview and security issues of the newly proposed approach. Section 4 reveals the results observed during the simulation. Section 5 analyzes the advantages of our approach and Section 6 concludes the work.

2. Related work

The cloud service provider should be able to provide the intended services and be able to secure itself from serious threats [17] such as RAS (Reliability, Availability and Security). A serious threat to cloud security is unauthorized access which can be avoided by Non- repudiation. DDoS filtering at network layer [18] reduces the attack packet rate and allows the HTTP requests for further processing. Intermediate nodes in the internet cloud identify the threat without transferring it to the protected server. Packet scoring and Confident Based Filtering [19] are used to identify and predict threats. Based on the score, packets are either allowed to access the server or filtered outside the network.

Availability [1] in cloud computing not only refers to the data in DC but also the resources. Timing faults [14] deal with two kinds of DoS (Denial of Service) attackers. In Resource DoS, resources of DC are depleted by the forged attack packets and are unable to serve legitimate requests. DDoS defense mechanism [15], like hop count filter, anomaly detectors, normal profile creation and attacker profile creation reduce false positives and false negatives thereby improve attacker detection schemes. Over Court Gateways [2] is a credit based system where well behaving users will gain credit points and ill behaving users will lose their credit points. Migration based response [3] relocates the physical host after detecting the attack. This migration approach is considered as a preventive action.

Low rate DDoS detection scheme [4], Cloud Protector [5] & Cloud Trace Back filter and Advanced Cloud Protection System [6] increased the security of

cloud resources. Handling DDoS [7] requires filtering the flooding attack and processing legitimate traffic. AID [8] is a complete self-defense system but the legitimate traffic is protected from server access. IP traceback [20, 21], Pushback [22] and Path Identification [23] schemes are effective with DoS attacks but not with DDoS attacks. Availability and privacy are serious issues [24] for the users of a cloud infrastructure. The authenticated users are given access and they are queued at the whitelist whereas the unauthenticated users are blacklisted and filtered at firewall [25].

Futuristic technology [9] reduces the IT services cost and helps in improving the availability, flexibility, reliability and throughput by reducing the processing overhead. RVWS [10] with dynamic attributes and stateful web services make the service available at any time. Resource provisioning scheme [11] should be able to keep track of available resources to satisfy the minimum requirements. In Agent based Bag of Task [16], the dynamic allocation along with rescheduling improves the resource utilization. Fuzzy pattern [12] to filter Botnets uses behavior based pattern to detect the Botnets attack. This achieves traffic reduction and eliminates false positives and false negatives. Policy based resource allocation [13] reveals that the resources can be allocated iff they are available. In the eventuality all the resources are occupied by the legitimate users and requests are kept waiting. They are rejected after timeout.

All the above defense and resource provisioning schemes tried to improve the availability by detecting threats and serve the intended clients with increased cost. The proposed approach is lightweight, cost effective with improved resource utilization and at the same time achieves its objective of all time availability.

3. Cloud DDoS attack protection – architecture and security requirements

3.1. CDAP architecture overview

Cloud DDoS Attack Protection (CDAP) nodes are the army nodes deployed between the client and the DataCenter, which form a ring by connecting with each other serially, shown in Fig. 1. The CDAP nodes could classify and identify the legitimate users from attackers. Legitimate clients behavior differs when they are either compromised or acquired by the attackers. The registered users are of two kinds, authenticated/valid user for the session and unauthenticated/registered but waiting for authentication for accessing server resource. The unregistered users are new users to the server. The IP addresses which are unavailable within the REGISTER_STATUS are considered to be unregistered users. Requests from unregistered users are sent to the authentication module where the legitimacy of the user has been intimated to the CDAP. The CDAP node generates and stores a session key which is shared with the user as well. The requests from the existing clients will be authenticated and session key is generated upon request for the server resource.

- When the number of requests to be handled by the CDAP server increases beyond the threshold value, the request is automatically forwarded to the other CDAP servers in the network.
- When the number of requests generated by the same user is more than *N* requests at a time, by verifying the REGISTER_STATUS, the particular user is blacklisted for that particular session and is recorded in the BLACKLIST_CLIENT table. Further requests from such users are blocked and redirected to the DUMP terminal outside the cloud.

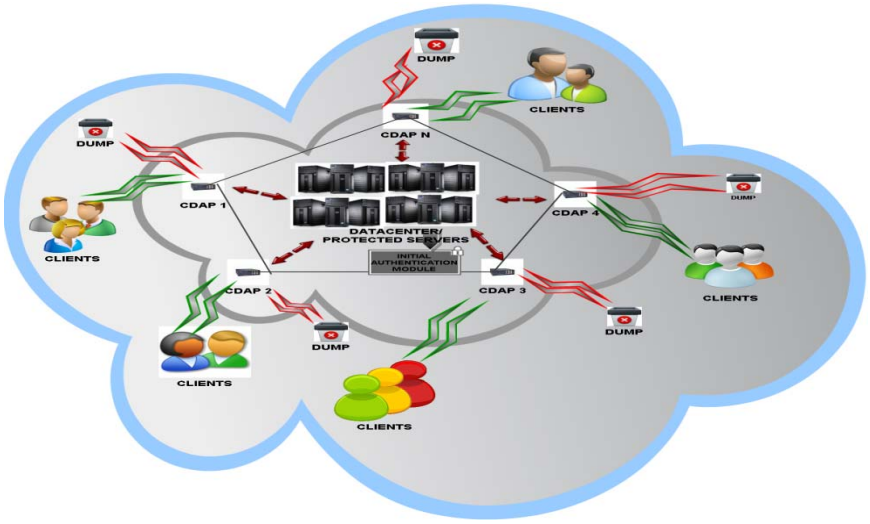

Fig.1. Architecture of CDAP

CDAP could either be a host or a dedicated VM in DataCenters. A DataCenter might have any number of CDAP hosts based on its processing capability and attack prone zones. A CDAP being implemented as a dedicated host has its own DUMP terminal which handles the requests from the attackers by ignoring it. The CDAP implemented as a VM in DataCenters however forwards the request to the DUMP terminal through the corresponding CDAP.

3.2. Detailed design of proposed architecture

The detailed working mechanism of the CDAP is depicted in Fig. 2.

[1] Interested clients request Name Resolution Server (NRS) for the address of protected server.
[2] Instead of protected server address, NRS responds with the CDAP address which is nearest to the requesting client.
[3] On acquiring the address, the client sends a request to CDAP.
[4] Based on the incoming IP behavior at REGISTER_STATUS, the client's request will be forwarded to server.
[5] Unregistered user will be registered only when the legitimate protocol is followed. At CDAP if REGISTER_STATUS table has any user with new registration and follows the legitimate protocol, it will be sent to

AUTHENTICATION module which adds the user to the network. The existing clients will be authenticated and a session key is generated upon request for server resource.

6 on successful registration, the CDAP will be notified about registration and forwarded to the requester who requests for new registration. If the existing users are authenticated successfully then the session key is generated and supplied to the requestor via CDAP, stores the session key.

7 Server responds to the clients request by forwarding the necessary details, session key, to army nodes.

8 CDAP redirects the session key details to the intended users.

9 At CDAP1, if the number of clients (registered) reach their threshold, then the request is forwarded to other CDAP and if other user (unregistered) has more than N, register the request in REGISTER_STATUS table within the same session. This particular user is blacklisted for that particular session in BLACKLIST_CLIENT table for initiating the DDoS attack. These requests are redirected to DUMP terminal, resides outside the cloud network.

10 In case of any CDAP failure, table contents will be forwarded to neighbor CDAP.

11 Any new users who needs to join in the network.

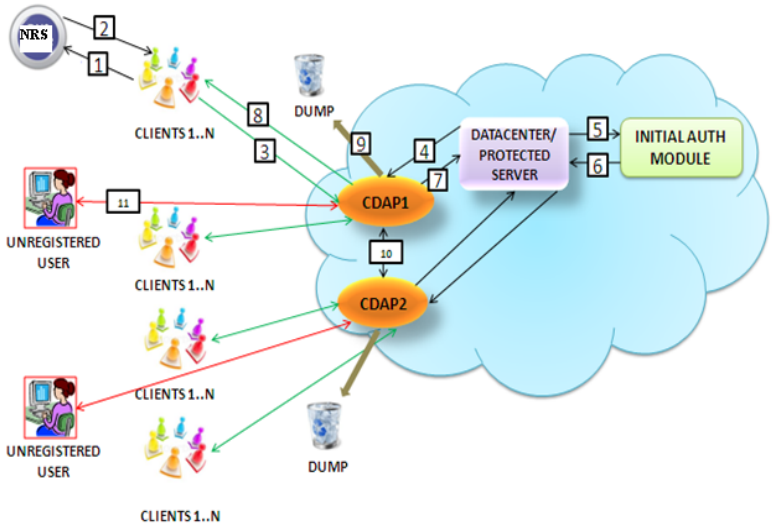

Fig. 2. Working mechanism of proposed architecture

3.2.1. CDAP logs

a. REGISTER_STATUS

The REGISTER_STATUS log keeps track of human users and eliminates the botnets. The log has a list of authenticated as well unauthenticated registered clients, the IP address of the requests from the unregistered user and the IP address of the neighbor CDAP as shown in Table.1. If the number of requests from the same unregistered user exceeds *N*, the user is identified as a DDoS attacker.

Table 1. REGISTER_STATUS log

Registered client (server access request)		Unregistered user (new registration request)	Neighbour CDAP request
Autheticated	Unauthenticated		
192.168.123.1	192.168.123.9	192.168.123.211	192.168.123.123
192.168.123.2	192.168.123.250	192.168.123.223	192.168.123.222
…	…	…	…

b. REQUEST_STATUS

The REQUEST_STATUS log has the client IP along with the request rate and its size, shown in Table 2. A dramatic increase in request rate within the REQUEST_STATUS blocks the IP.

Table 2. REQUEST_STATUS log

Client IP	Request rate (max request per hour=3600)	Request size (bytes)
192.168.123.1	360	3000
192.168.123.2	1000	3000
…	…	…

c. Session table

Session table helps in identifying the number of users currently connected to the particular CDAP. The resources are allocated to users only if the request rate and request size is below the threshold, which is identified as a legitimate request arrival. The detailed description can be found in [28]. The intention of the CDAP approach is to prevent the attacker's entry into cloud network. The Distributed attacker's strength has to be weakened by splitting the huge attacker group and preventing attacker entry. This eventually leads to entry of the legitimate request into the cloud network. By doing so, the detection scheme should be light-weight and should not impose any heavy computational overhead. The distributed attackers are to be treated as distributed. The small groups of attackers are easily detected there and deactivated using the BLACKLIST_CLIENT log.

d. BLACKLIST_CLIENTS

Requesting IP addresses are classified into four categories as shown in Table 3 and these details are forwarded to CDAP. The identified attackers are logged in BLACKLIST_CLIENTS. The Client IP when found in the log, the corresponding request is restricted from reaching the server.

Table 3. BLACKLIST_CLIENTS log

S. No	Class A	Class B	Class C	Dead address	Others/ Neighbour CDAP black listed
1	123.181.12.232		192.168.123.223		
…	…	…	…	…	…

Neighbour CDAP details in Tables 1 and 3 are periodically exchanged to serve legitimate connections to the failed CDAP in worst case scenario. On sufficient CDAP deployment this field can be deactivated. This field is added to improve availability even at the time of CDAP failure in worst cases and the CDAP remains active at all time, if CDAP is configured with logs. Here, the authentication module shows the session key generation and exchange.

3.3. CDAP's security requirement

3.3.1. Access control to server

The clients send the request to the DataCenters via CDAP. If the incoming user IP is found in BLACKLIST_CLIENT log, the requests are forwarded to DUMP else forwarded to the DC and a session key is generated. The session key is sent to the user via CDAP and the CDAP stores the session key. The successive requests will be validated based on the session key at CDAP. The session key status is updated based on the legitimate user actions at the time of session expiration.

3.3.2. Secured data exchange

On successful session key generation at Datacenter/ protected server, CDAP validates further user's requests based on the session key. This enhances the security by allowing only the legitimate clients whose session key matches with the key stored at CDAP which was generated already at the Authentication module of Datacenter. This also saves time by validating a session key instead of monitoring with the special anomaly detector.

3.3.3. Service restriction

The process of periodic status monitoring about the number of active clients is sent to each CDAP. Whenever a client requests the CDAP (sub server), the client's IP is logged in REGISTER_STATUS log and their requests size and request rate (significant characteristic to launch DDoS) are recorded at REQUEST_STATUS log. Whenever the attacker profile or deviation in legitimate user protocol is identified, the particular client's IP is moved to the BLACKLIST_CLIENTS log. The clients IP in BLACKLIST_CLIENTS log are Ingress filtered (not allowed) to deny access to Datacenter resources. If the incoming request rate or request size of any client exceeds the legitimate profile, the access is restricted and Ingress Filtered.

3.3.4. Traffic control

The communication of the Periodical status of the number of active clients is sent to each CDAP. In case the CDAP (CDAP 1) is filled with its maximum number of legitimate sessions, the further incoming legitimate clients are forwarded to an under loaded neighbor CDAP (CDAP 2). From then on, the legitimate client is a part of CDAP 2. If the incoming requester deviates from the legitimate client protocol, the Client IP will be sent to BLACKLIST_CLIENT log. This proposed

scheme shows it has built-in Load Balancing based on the virtual firewall architecture (CDAP surrounds Datacenters).

3.4. Pseudocode of CDAP approach

The pseudocode of CDAP approach is as following.

If (Clients require Server Resources)
Acquire IP from NRS
Send credentials to CDAP
At CDAP

Step 1: Identification of incoming client and if botnet- service restricted

If (Client IP address is not found in ***REGISTER_STATUS.unregistered*** *(incoming Client IP))*
ADD Client IP to ***REGISTER_STATUS.unregistered*** *(Client IP)*
*Else If (****REGISTER_STATUS.unregistered*** *(same Client IP)>N [within session TIMEOUT period])*
MOVE Client IP to ***BLACKLIST_CLIENT*** *table*
BLACKLISTED users are INGRESS filtered at CLOUD network.
DDoS INITIATION alert (botnet).
Else
Forward Client credentials to PROTECTED SERVER for registration
*Else (****REGISTER_STATUS.registered.unauthenticated****(Client IP)) //already registered*
Forward Client credentials to PROTECTED SERVER for authentication

Step 2: monitoring the request rate. If abnormal request rate-attacker found

If (for any Client IP(REQUEST_STATUS.request rate<= X && REQUEST_STATUS.request size <=Y))
Forward Client credentials to PROTECTED SERVER/ DATACENTER.
Else
MOVE Client IP to ***BLACKLIST_CLIENT*** *table.*
BLACKLISTED users are INGRESS filtered at CLOUD network
DDoS INITIATION alert (human attacker group)

AT SERVER:

Step3: Only Registered users or unregistered user who follows the protocol of N,X,Y

If (authentication success)
Session key generated and stored at CDAP SESSION TABLE and sent to Clients by CDAP.
If (CDAP ***SESSION TABLE*** *reaches its max)*
The log details forwarded to neighbor CDAP node.
Else
Failure notification sent to CDAP.

AT CDAP: - On receiving response from SERVER:

Step 4: Access to server resource and Data processing

If (session key generated)
Update ***REGISTER_STATUS.unregistered (Client IP)*** *to* ***REGISTER_STATUS.registered.authenticated (Client IP)***
Else
Packets forwarded to DUMP terminal by disallowing to reach at server.

4. Simulation results

The simulation duration is one hour and the results are proven by the expenditure incurred at the DataCenter due to attackers (with and without CDAP).

4.1. Response time

Fig. 3 shows the response time oscillation in the presence and absence of CDAP. Increasing the number of DataCenters reduces the average response time. However the Datacenter with CDAP almost behaves normally in all the scenarios irrespective the number of DC.

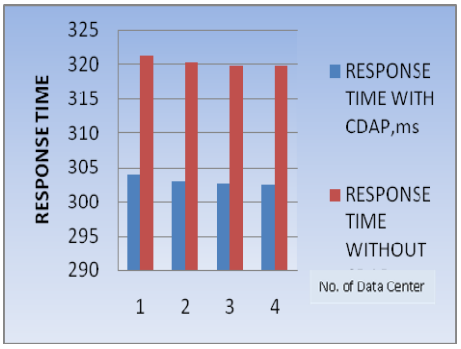

Fig. 3. Response time

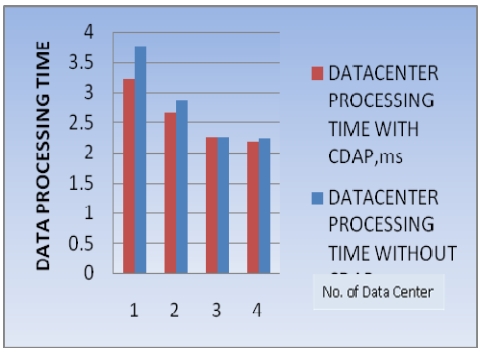

Fig. 4. Data processing time

4.2. Datacenter processing time

Fig. 4 shows that data processing time can be improved by increasing the DataCenters. Attack on a single victim DataCenter without CDAP increases average processing time. Also it reduces when number of DataCenters increase for processing the request but DataCenters with CDAP face less processing as the attackers are ejected out by CDAP. DataCenters with CDAP support VM time shared scheduling. Also it supports VM migration which involves multitasking and resembles time sharing systems.

4.3. Resource usage cost (VM + Bandwidth + Ram + Storage)

Fig. 5 shows the resource depletion by the attacker (without CDAP) and resource protection (with CDAP). Since attackers are detected and filtered at CDAP, it saves on resource cost. The Resource usage cost includes the cost of utilizing the VM, Bandwidth, RAM, and Storage. This drastic difference in cost for the sample scenario with different number of DataCenters shows the severe effect of DDoS. But the same scenario when executed with CDAP shows its extreme significance. DataCenters with CDAP eliminates the attackers and allows the client's request within the legitimate request size.

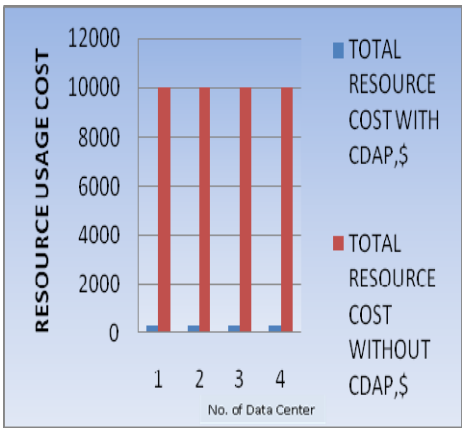

Fig. 5. Total resource usage cost

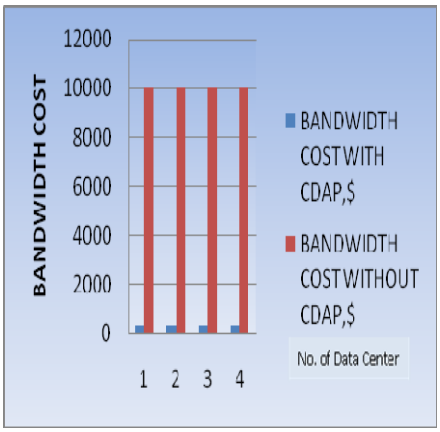

Fig. 6. Bandwidth cost

4.4. Bandwidth resource cost/ Data transfer cost

Fig. 6 shows DataCenter without CDAP, losing bandwidth resources when severely attacked by DDoS attackers. The bandwidth resource cost shows that the attacker's request load entered the cloud network without CDAP whereas the requests are filtered and redirected to DUMP when DataCenters were deployed with CDAP.

4.5. VM usage cost

Fig. 7 shows increase in DataCenters, lead to increase in cost because VM migration resembles time sharing systems. Scalable /elastic resources are allotted, suspended and resumed based on the client's priority which in turn increases time and cost.

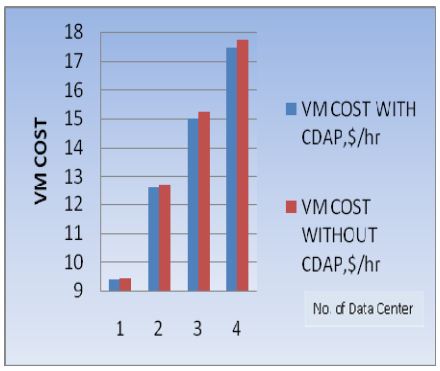

Fig. 7. DataCenter load

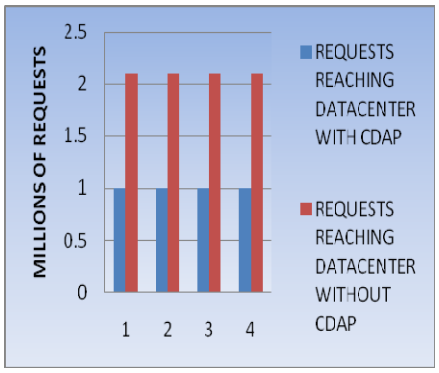

Fig. 8. VM usage cost

DC without CDAP takes around 2.5 times to simulate the same scenario simulated under DC with CDAP. This causes increase in cost due to VM migration (leads to increases in time) because VM usage cost is calculated in $/hr. Increase in DataCenters without CDAP increases the VM resource cost.

4.6. Load reaches datacenter

The requests reaching DataCenter under DDoS have been recorded and shown in Fig. 8. Request Load is shared based on the number of DataCenters. Also it proves that our approach is highly feasible to implement which helps in authenticating the legitimate users and identifying attackers at attack initiation.

5. Analysis of CDAP approach

The main difference between the legitimate request and the DDoS attacker request is the varied traffic pattern [27]. The attacker may also try to impersonate a legitimate request, but the CDAP's log helps in identifying such attacks by tracking user details in REGISTER_STATUS table.

DC without CDAP. Virtual Machines are allocated to the incoming combination of traffic at high rate because the large number of requests populated by the attackers reaches the DC and response time slows down. After a period of time the DC responds poorly and finally it may not respond at all.

DC with CDAP. CDAP acts as a firewall by validating the session. VM ID is the identity assigned to VM to service the incoming request and to evaluate the total number of VM that completes the incoming request processing. Number of VM Resources Allocated –Number of times the particular VM is involved in allocating VM resource to complete the incoming request processing. This solely depends on the VM allocation policy deployed in request processing. The multitasking capability and 75 simultaneous request processing initiated at VM allocation in DC is shown in Figs 9-12.

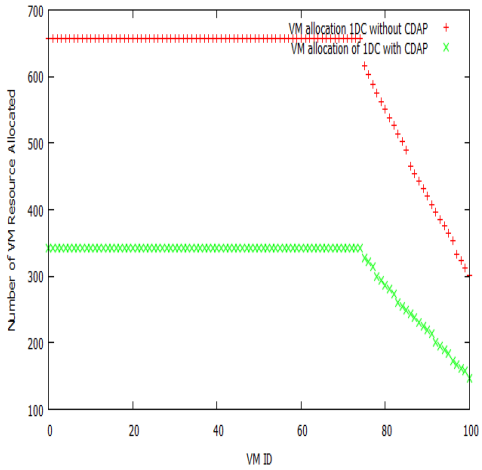

Fig. 9. VM Allocation in one DataCenter

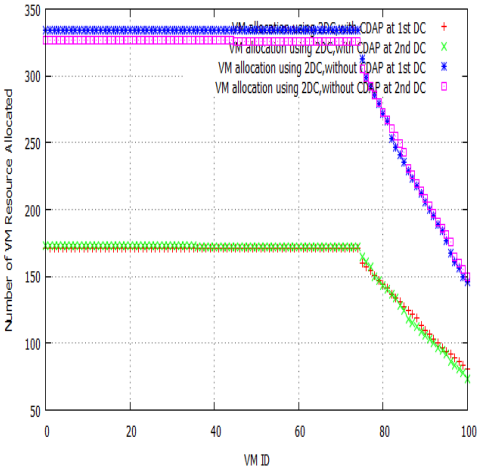

Fig. 10. VM Allocation in two DataCenters

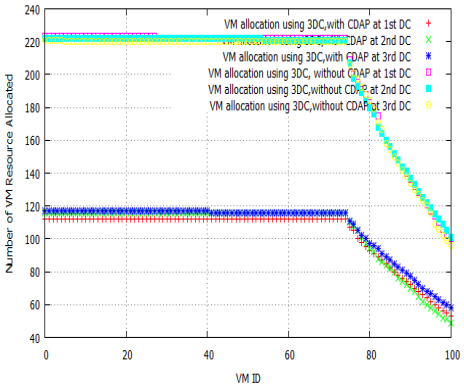

Fig. 11. VM allocation in three DataCenters

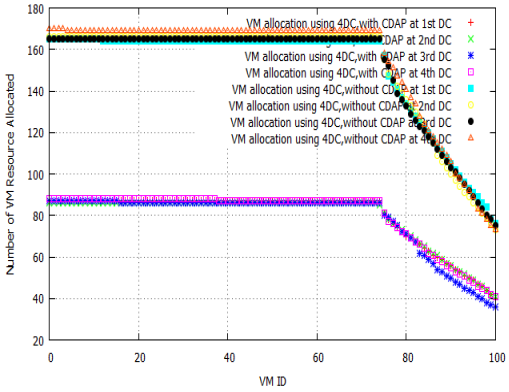

Fig. 12. VM allocation in four DataCenters

Table 10 lists comparison between the maximum numbers of VM processing resources allocated to the Datacenters for processing the same number of incoming requests with and without CDAP.

Table 10. Maximum VM resource allocations on demand at DC with and without CDAP

Total number of DC	VM allocation at 1st DC		VM allocation at 2nd DC		VM allocation at 3rd DC		VM allocation at 4th DC	
	Without CDAP	With CDAP	Without CDAP	With CDAP	Without CDAP	With CDAP	Without CDAP	With CDAP
1	658	343	–	–	–	–	–	–
2	334	171	327	173	–	–	–	–
3	223	112	222	116	221	117	–	–
4	165	87	166	86	165	87	170	88

Table 11 shows that minimum VMs Allocated at DC in Figs 11, 12, 13, 14. The notable point is that VM Resource allocation decreases when the number of DC increases. The VMs are allocated for incoming requests and after their execution they are returned back for other requests processing. The DC with CDAP will allow only legitimate users and finish the submitted task quickly but the number of VM allocated on demand to DC without CDAP acquires the resources (VM) and the legitimate users are disallowed and delayed because of DDoS flood attack, as shown in Tables 10 and 11.

Table 11. Minimum VM resource allocations on demand at DC with and without CDAP

Total number of DC	VM allocation at 1st DC		VM allocation at 2nd DC		VM allocation at 3rd DC		VM allocation at 4th DC	
	Without CDAP	With CDAP	Without CDAP	With CDAP	Without CDAP	With CDAP	Without CDAP	With CDAP
1	302	147	–	–	–	–	–	–
2	146	81	150	73	–	–	–	–
3	100	53	101	49	96	58	–	–
4	76	40	73	41	75	36	73	41

5.1. Availability ratio

In worst case scenario, the failed CDAP passes the requests to active CDAPs and balances the load generated by DDoS attackers towards DataCenters. Presence of other active CDAP nodes will improve the availability, by sharing the incoming requests among them. At any time t, the server availability depends on the number of active CDAP nodes as shown in the equation (1)

$$A(t) = \sum_{i=1}^{k} \frac{\text{CDAP}(i)}{n} \tag{1}$$

where $A(t)$ is the availability of protected server at time t, n is the number of CDAP nodes that surrounds the protected server initially, k being number of Active CDAPs, and CDAP(i) being the active CDAP. Datacenter processing capability and the availability is represented in equation (2).

$$\text{AVAIL\_DDoS}(t) = n - \sum_{i=1}^{l} \text{CDAP}_{\text{in}}(i) \tag{2}$$

where $\text{AVAIL\_DDoS}(t)$ = Availability of protected server at time t during DDoS; l = number of Active CDAP; CDAP_{in} = Inactive CDAP.

5.2. Percent of Availability during DDOS

The availability of the server at the time of DDoS attack is the percentage of ratio between Availability of protected server at time t during $DDoS$ has given in equation (2) and the number of CDAP nodes that surround the protected server initially as shown in equation (3)

$$\text{A\_DDoS}(\%) = \left[\left(\frac{\text{AVAIL\_DDoS}}{n} \right) \times 100 \right] \% \tag{3}$$

where n is the number of CDAP nodes that surround the protected server initially, A_DDoS(%) being the percentage of server availability to clients during DDoS.

6. Conclusion and future enhancements

This architecture is less cost consuming because the server still serves the clients and makes the server available to its intended clients even during a DDoS attack at CDAP (sub server). Treating the DDoS attackers with distributed CDAP, the interlinked army nodes acts as a virtual firewall to DataCenters, has a positive effect in mitigating the attack and thus saving resources for the service provider, clients with minimal response time.

This proposed scheme can be applicable to Datacenters which are prone to DDoS attacks. This scheme shows its efficiency in identifying the attackers at the initial stage of DDoS launch and eliminates them immediately. The strategy “Tit for Tat” means that the distributed attackers are treated by distributed CDAPs at DC

without requiring any other DC to be involved in load balancing. Another strategy "Divide and conquer" is used so that the distributed attackers' strength is retarded by dividing the attacker group by reaching towards different CDAP and the CDAP has its own processing capability to identify the threat initiators. Finally we conclude by proposing the "light weight detection scheme" for "light weight computing". Unlike any other distributed computing, cloud computing only requires web interface, so the proposed scheme will work better than the probabilistic detection schemes.

Acknowledgements: We would like to thank Dr. N.Ch.S.N.Iyengar, who motivated and guided us in the right direction to accomplish the task. We would also like to thank the School of Information Technology and the management of VIT University for their support and motivation for encouraging us to pursue the proposed system.

References

1. Zissis, D., D. Lekkas. Addressing Cloud Computing Security Issues. – Future Generation Computer Systems, Vol. **28**, 2012, 583-592.
2. Du, P., A. Nakao. OverCourt: DDoS Mitigation through Credit-Based Traffic Segregation and Path Migration. – Computer Communications, Vol. **33**, 15 December 2010, No 18, 2164-2175.
3. Lent, R. Evaluating a Migration-Based Response to DoS Attacks in a System of Distributed Auctions. – Computers & Security, Vol. **31**, May 2012, No 3, 327-343.
4. Liu, Xiao-Ming, Gong Cheng, Miao Zhang, Shou-Shan Luo. On a Novel Pattern of Distributed Low-Rate Denial of Service Attacks. – The Journal of China Universities of Posts and Telecommunications, Vol. **18**, December 2011, No 2, 113-118.
5. Chonka, A., Yang Xiang, Wanlei Zhou, Alessio Bonti. Cloud Security Defense to Protect Cloud Computing Against HTTP-DoS and XML-DoS Attacks. – Journal of Network and Computer Applications, Vol. **34**, July 2011, No 4, 1097-1107.
6. Lombardi, F., R. Di Pietro. Secure Virtualization for Cloud Computing. – Journal of Network and Computer Applications, Vol. **34**, July 2011, No 4, 1113-1122.
7. Janczewski, Dr L. J., D. Reamer, J. Brendel. Handling Distributed Denial-of-Service Attacks. – Information Security Technical Report, Vol. **6**, 2001, 37-44.
8. Chen, Shigang, Yibei Ling, Randy Chow, Ye Xia. AID: A Global Anti-DoS Service. – Computer Networks, 2007, 4252-4269.
9. Goscinski, A., M. Brock. Toward Dynamic and Attribute Based Publication, Discovery and Selection for Cloud Computing. – Future Generation Computer Systems, Vol. **26**, 2010, 947-970.
10. Iqbal, W., M. N. Dailey, D. Carrera, P. Janecek. Adaptive Resource Provisioning for Read Intensive Multi-Tier Applications in the Cloud. – Future Generation Computer Systems, Vol. **27**, 2011, 871-879.
11. Vecchiola, C., R. N. Calheiros, D. Karunamoorthy, Rajkumar Buyya. Deadline-Driven Provisioning of Resources for Scientific Applications in Hybrid Clouds with Aneka. – Future Generation Computer Systems, Vol. **28**, 2012, 58-65.
12. Wang, Kuochen, Chun-Ying Huang, Shang-Jyh Lin, Ying-Dar Lin. A Fuzzy Pattern-Based Filtering Algorithm for Botnet Detection. – Computer Networks, Vol. **55**, 2011, 3275-3286.
13. Nathani, Amit, Sanjay Chaudhary, Gaurav Somani. Policy Based Resource Allocation in IaaS Cloud. – Future Generation Computer Systems, Vol. **28**, 2012, 94-103.
14. Arshad, Junaid, Paul Townend, Jie Xu. A Novel Intrusion Severity Analysis Approach for Clouds. – Future Generation Computer Systems, 16 August 2011, ISSN 0167-739X. **10.1016/j.future.2011.08.009.**

15. V a r a l a k s h m i, P., S. T h a m a r a i S e l v i. Thwarting DDoS Attacks in Grid Using Information Divergence. – Future Generation Computer Systems, 18 November 2011. ISSN 0167-739X.
10.1016/j.future. 2011.10. 012.
16. G u t i e r r e z-G a r c i a, J. O., K. M. S i m. A Family of Heuristics for Agent-Based Elastic Cloud Bag-of-Tasks Concurrent Scheduling. – Future Generation Computer Systems. ISSN 0167-739X.
10.1016/j.future.2012.01.005.
17. S a b a h i, F a r z a d. Cloud Computing Security Threats and Responses. – In: Proc. of IEEE 3rd International Conference on Communication Software and Networks, 2011, 245-249.
18. D u, P i n g, A k h i r o N a k a o. DDoS Defense as a Network Service. – In: Proc. of IEEE/IFIP Network Operations and Management Symposium, 2010, 894-897.
19. C h e n, Q i, W e n m i n L i n, W a n c h u n D o u, S h u i Y u. CBF: A Packet Filtering Method for DDoS Attack Defense in Cloud Environment. – In: Proc. of 9th IEEE International Conference on Dependable, Autonomic and Secure Computing, 2011, 427-434.
20. L e e, H e e j o, K i h o n g P a r k. On the Effectiveness of Probabilistic Packet Marking for IP Traceback under Denial of Service Attack. – In: Proc. of INFOCOM, 2001, 338-347.
21. S a v a g e, S., W e t h e r a l l, D. K a r l i n, A. T o m, A. T o m. Practical Network Support for IP Traceback. – SIGCOMM, Vol. **30**, 2000, No 4, 295-306.
22. I o a n n i d i s, J., S. M. B e l l o v i n. Implementing Pushback: Router-Based Defense Against DDoS Attacks. – In: Proc. of Network and Distributed System Security Symposium, Vol. **2**, February 2002.
23. Y a a r, A., A. P e r r i g, D. S o n g. Pi: A Path Identification Mechanism to Defend against DDoS Attacks. – In: Proc. of IEEE Symposium on Security and Privacy, 2003, 93-107.
24. J o s h i, K. R. G. B u n k e r, F. J a h a n i a n, A. P. A. v a n M o o r s e l, J. W e i n m a n. Dependability in the Cloud: Challenges and Opportunities. – In: Proc. of 2009 IEEE/IFIP International Conference on Dependable Systems and Networks, DSN, 2009, Estoril, Lisbon, Portugal, 29 June-2 July 2009, 103-104.
25. S q a l l i, M. H., F. A l-H a i d a r i, K. S a l a h. EDoS-Shield – A Two-Steps Mitigation Technique Against EDoS Attacks in Cloud Computing. – In: Proc. of 4th IEEE International Conference on Utility and Cloud Computing, December 2011, 49-56.
26. J a i n, P r i t e s h, D h e e r a j R a n e, S h y a m P a t i d a r. A Survey and Analysis of Cloud Model-Based Security for Computing Secure Cloud Bursting and Aggregation in Renal Environment. – In: Proc. of World Congress on Information and Communication Technologies, December 2011, 456-461.
27. J u n g, J., B. K r i s h n a m u r t h y, M. R a b i n o v i c h. Flash Crowds and Denial of Service Attacks: Characterization and Implications for CDNs and Web Sites. – In: Proc. of 11th International Conference on World Wide Web (WWW'02), ACM, 2002, 293-304.
28. B u y y a, R a j k u m a r, R a j i v R a n j a n, R. N. C a l h e i r o s. Modeling and Simulation of Scalable Cloud Computing Environments and the CloudSim Toolkit: Challenges and Opportunities. – In: Proc. of 7th High Performance Computing and Simulation Conference (HPCS), 21-24 June 2009.