Computer Networks 109 (2016) 157–171

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

DDoS attacks in cloud computing: Collateral damage to non-targets

Gaurav Somani[a,b,*], Manoj Singh Gaur[b], Dheeraj Sanghi[c], Mauro Conti[d]

[a] Central University of Rajasthan, Ajmer, India
[b] Malaviya National Institute of Technology, Jaipur, India
[c] Indian Institute of Technology, Kanpur, India
[d] University of Padua, Padua, Italy

ARTICLE INFO

Article history:
Received 16 October 2015
Revised 13 March 2016
Accepted 30 March 2016
Available online 8 April 2016

Keywords:
Cloud computing
Distributed denial of service (DDoS)
Economic denial of sustainability (EDoS)

ABSTRACT

The effects of distributed denial-of-service (DDoS) attacks on cloud computing are not very similar to those in traditional "fixed" on-premise infrastructure. In the context of DDoS attacks in multi-tenant clouds, we argue that, instead of just the victim server, multiple other stakeholders are also involved. Some of these important stakeholders are co-hosted virtual servers, physical servers, network resources, and cloud service providers. In this paper, we show through system analysis, experiments, and simulations that these stakeholders are collaterally affected, even though they are not the real targets of the attack. Damages/effects to these stakeholders include performance interference, web service performance, resource race, indirect EDoS (economic denial of sustainability), service downtime, and business losses. The result of our cloud-scale experiment revealed that overall energy consumption and the number of VM migrations are adversely affected owing to DDoS/EDoS attacks. To the best of our knowledge, this work is the first novel contribution in regard to the effect characterization on non-targets in the cloud computing space. We make an effort to identify the targets of these effects and their origins, such as auto-scaling, multi-tenancy, and accounting in the cloud. We argue that there is an immense need to relook at the DDoS solutions in the cloud space where efforts are needed to minimize these effects. Finally, we have identified the detailed requirements of mitigation solutions to DDoS attacks in the cloud with an aim to minimize these effects. We provide an ideal solution design by taking characterization outcomes as important building blocks.

1. Introduction

Distributed denial-of-service (DDoS) attacks have drawn extensive attention in cyberspace in the last few years. According to the survey conducted by Kaspersky, one out of five businesses has faced a DDoS attack in 2014 [1]. Attacks ranging beyond 100 Gbps have highly increased and harmed organizations in multiple ways including business reputation, business losses, and downtime [2]. DDoS attack, and its fatal version, EDoS (economic denial-of-sustainability) attack, in the cloud has gained much attention from the security research community. In the report in [3], the authors anticipated a shift in DDoS attackers' target to cloud-based services. The multiple incidents in 2014 and 2015 are strong evidences for the predictions in this report. According to the report in [4] from Verisign iDefense Security Intelligence Services, the cloud and SaaS (software-as-a-service) sectors are the most attacked targets of recent DDoS attacks. This report also revealed that more than one-third of DDoS mitigations included cloud-based services under attack. There have been four important attack incidents in the recent past, which have attracted a great deal of discussions in the security research community. Attacks on Microsoft and Sony gaming servers (hosted in the cloud) by Lizard Squad were the first incident. Similarly, the servers of cloud service provider Rackspace were targeted by a massive DDoS attack. The third incident was also fatal, where Amazon EC2 servers were attacked [3]. More recent fatal attacks included one of the cloud hosting providers, Linode, which was targeted by a number of DDoS attacks over a week across its worldwide data centers [5]. One more factor that is important to appreciate the contribution of our paper is "economic losses." As per the report of Kaspersky Labs [1], the average financial damage caused by a single DDoS attack is $444,000. Similarly, the Q1 data for 2015 by Neustar revealed that the average losses resulting from a DDoS shutdown were as high as £100,000/h [6]. Another DDoS victim, Greatfire.org, which is based on the Amazon EC2 cloud, faced a heavy DDoS attack in March 2015, costing it a huge bill of $30,000 daily [7]. DDoS attacks in the cloud require efforts at multiple levels for effective solutions with a minimum outage. Recent works

* Corresponding author.
E-mail address: gaurav@curaj.ac.in (G. Somani).

http://dx.doi.org/10.1016/j.comnet.2016.03.022
1389-1286/

have characterized the DDoS attacks, citing them as either EDoS attacks [8], fraudulent resource consumption (FRC) attacks [9,10], or fraudulent energy consumption attacks [11,12]. In this type of DDoS attack, the major aim of the attackers is to attack the sustainability aspects of the victim server. Economic aspects are affected because of the high resource and energy usage, and the resultant resource addition and plugging, thus generating heavy bills owing to the "pay-as-you-go" billing method. Mostly, DDoS mitigation methods are employed at the application layer of the server, where anomaly detection is performed on incoming web traffic [13–18]. These methods include preventive approaches [19–22] and detection approaches [9,23,24]. There are many other methods that employ network-level DDoS mitigation based on traffic monitoring at additional overlay servers and routers [25–27]. It is well established that the consequences of the attack will affect the target server and the services offered. We argue that this is not a correct and complete evaluation of attack consequences with servers hosted in infrastructure clouds. An infrastructure cloud will always have multiple "multi-tenant" physical servers hosting a large number of virtual machines (VMs) sharing various resources using different techniques of resource multiplexing and sharing. The control of each VM would belong to its owner, and it should ideally be isolated from other resource-sharing components. However, the multi-tenant and collaborative nature of the cloud makes a large difference in characterizing DDoS attacks as compared to traditional infrastructures. This work is the first novel contribution in this area to characterize and show the various effects on the stakeholders. These effects include both direct and indirect impacts while a DDoS attack is occurring. An effort has also been made to provide a guideline to distinguish, measure, and minimize these effects, as these effects would change the factors governing cloud pricing, IT chargeback, and dispute resolutions. This work is an extension of our work published in [28] and provides a comprehensive analysis of the various aspects of DDoS attacks in cloud computing and solutions.

This paper is organized as follows: Section 2 describes DDoS attacks in the cloud. Section 3 presents a system model of cloud infrastructure to support the arguments presented in this paper. We list a detailed set of requirements for planned experiments in Section 4. Experiments and results are discussed in Section 5. Quantification of the results and their applicability contexts are discussed in Section 6. Section 7 is devoted to identifying the requirements to mitigate DDoS attacks in cloud computing. Finally, Section 8 and Section 9 present the related works and the conclusion, respectively.

2. DDoS/EDoS attack in the cloud

A DDoS attack targets a victim server by sending it a large number of service requests from a group of distributed clients/bots. In an infrastructure cloud, the attack scenario would be similar to the one shown in Fig. 1. An infrastructure cloud will have multiple high-capacity physical servers hosting VMs to meet the business objectives including return on investment and better hardware utilization. This cloud has a high-speed network to connect these servers to support applications and processes such as live VM migration. Usually, a cloud will have a queue of incoming VMs that will be placed on the servers. The cloud as a resource manager has multiple activities to perform, including VM placement, resource allocation, load balancing, accounting, and billing.

For our discussion, let us concentrate on physical server 3, which hosts four VMs in Fig. 1. VM1 is being targeted by a DDoS attack. A DDoS attack is usually achieved by targeting one or more of the basic server resources such as the CPU, memory, disk, bandwidth, number of TCP connections, open files, etc. It is important to note that the attack mechanism from an attacker's perspective

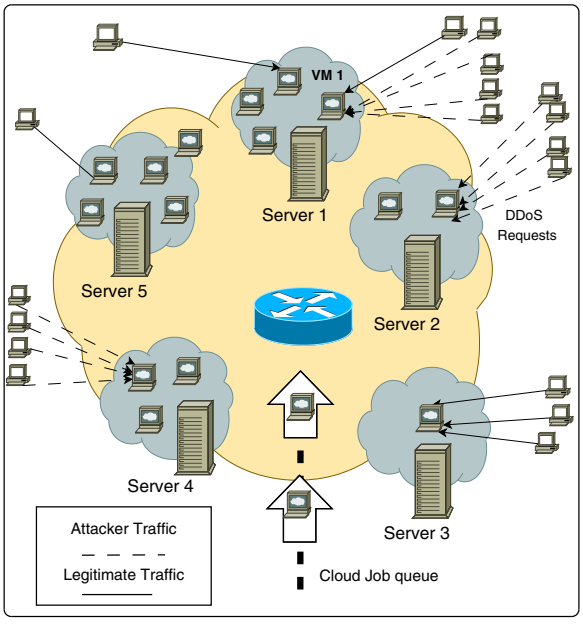


Fig. 1. DDoS scenario and various stakeholders.

will mostly remain the same for both on-premise fixed infrastructure and scalable cloud infrastructure. This makes it easy for an attacker to invest resources using the same attack strategies. However, for mitigation systems, it becomes a different task altogether. Infrastructure clouds are mostly appreciated for their capability to shrink and expand resources by offering on-demand computing facility. Most of these basic resources are being shrunk and expanded by cloud service providers using virtualization and auto-scaling mechanisms [29–32]. This would help the hosted VM avoid reaching a "denial-of-service" (DoS) state by increasing the resources to cope with the increasing demand. Even if the resources are exhausted in server 3, the DoS state may not be reached as the service running on VM1 can be scaled up vertically or horizontally on some other server, say, server 1, by creating another VM instance.

Theoretically, this may be done for infinite resources and instances owing to the availability of profound resources on the cloud. Obviously, these shrinking and expansion are linked to accounting and "pay-as-you-go" billing, attacking the economic sustainability of the VM owner. Practically, this attack may result in DoS as the maximum allowed resources to a VM owner cannot be infinite. These intermediate steps of continuous resource acquisition are important reasons behind EDoS attacks, which may or may not converge to a DDoS attack. This can be effectively understood by the resource allocation charts in Fig. 2. A web server instance (instance 1) has been started on a physical server in a cloud with basic resources of one unit. For simplicity, this one unit of resources represents a set of basic resources needed for one instance of a web server. As the web server is a virtualized server, the total resources are assumed to be 20 units, where four units are always reserved for the hypervisor or host domain and eight units are already occupied by other VMs. Hence, there are only seven units of idle resources left that are available to all VMs to acquire when needed. The auto-scaling decisions represented on the x-axis are taken by an algorithm for keeping track of VM resource utilization and other metrics (similar to the scheme in [33]). While the attack is being applied, each of these decisions would always result in "expansion."

As shown, the attack would consistently stress upon one or more of the basic resources and trigger the acquisition of all seven units of idle resources. Once this state is reached, there are

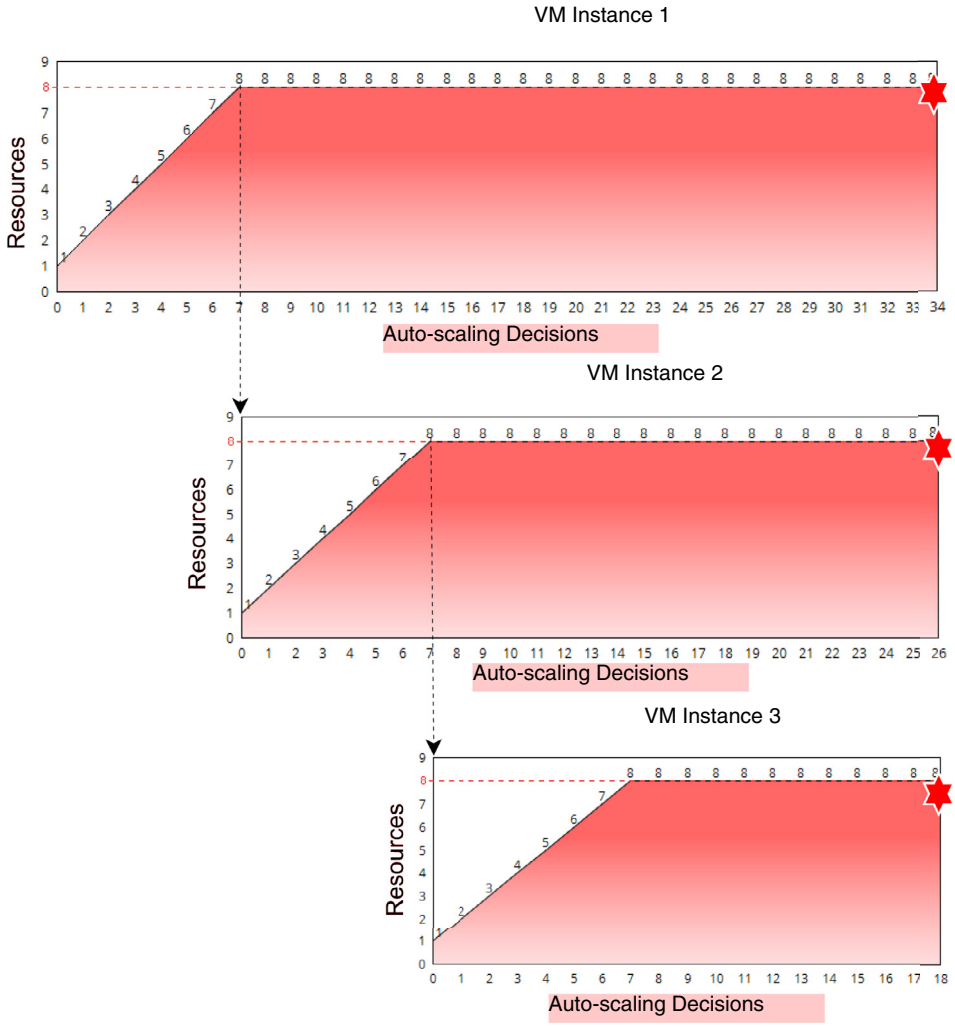


Fig. 2. EDoS and DDoS in the cloud.

no more idle resources available. Now, the auto-scaling algorithm would start another VM instance on the same server or some other server in the cloud. The location of the new instance creation is dependent upon various factors including candidate server identification, VM placement algorithm, future resource requirements, etc. Once another instance is started, there will be two parallel instances that are serving the incoming requests for the same service. Assuming that there is no DDoS mitigation service in place as the attack is being continued, this will also exhaust the resources available for the newly created instance, resulting in the creation of more and more instances. Theoretically, it may eat up all the resources of the cloud or all the resources allocated to the victim VM's owner. Finally, the web server in consideration would reach a state where service denial would happen (represented by a red star in Fig. 2).

In practice, a VM may not choose a model based on this discussion. Instead of creating more instances, it may choose migration to another server with more idle resources or a hybrid strategy combining all these approaches. Even in this case, the convergence from EDoS to DDoS will follow the same path. There are two matrices that are important to extend our discussion. Both these matrices are used in the next section for the development of the system model.

1. **Time to reach DoS**: The time required to reach a DDoS attack in the cloud will be higher than that in a traditional infrastructure (as there is only EDoS between auto-scaling decisions 1 and 7, and the service may be able to serve the requests).
2. **Attacker target**: If the attacker's aim is not toward service denial, it may send requests at a lower rate to realize the EDoS attack, which would economically harm but would not converge to DDoS.

3. System model

Considering an infrastructure cloud C that has P_n physical servers (similar to Fig. 1), we can represent each individual server as

$$P_i, i = 1, 2, ...n. \tag{1}$$

Similarly, the set of VMs that will actually run on these machines is V_m and these VMs are represented as

$$V_j, j = 1, 2, ...m. \tag{2}$$

Each machine P_i may have p resource types available. Similarly, a VM, V_j, may require q resource types using virtualization. Therefore, a physical server's resources will be

$$P_{ik}, k = 1, 2, ...p \text{ resource types..} \tag{3}$$

Similarly, a VM's resources would be represented as

$$V_{jl}, l = 1, 2, ...q \text{ resource types.} \tag{4}$$

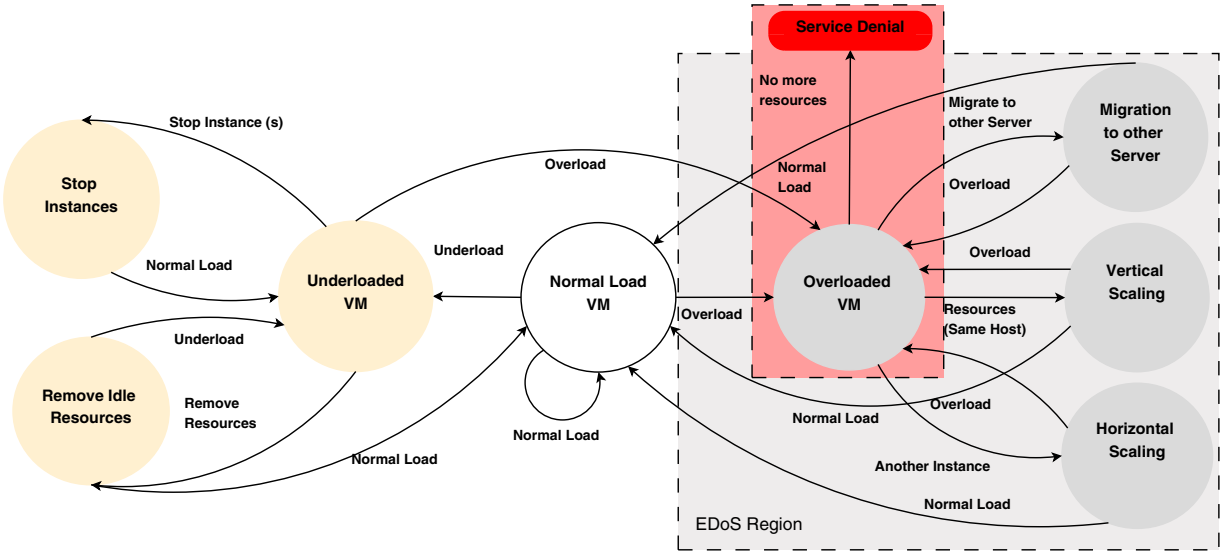


Fig. 3. State transition in cloud auto-scaling.

Usually, resource types are the CPUs (C), memory (M), disk space (D), and bandwidth (B). Additional resources can be added as per need. Therefore, a host P_i will have

$$P_{i1} = C_i, \quad P_{i2} = M_i, \quad P_{i3} = D_i, \text{ and } \quad P_{i4} = B_i. \tag{5}$$

Similarly, a VM V_j will have

$$V_{j1} = C_j \quad V_{j2} = M_j \quad V_{j3} = D_j, \text{ and } \quad V_{j4} = B_j. \tag{6}$$

The total resource capacity of a physical host would be the total resources available at the host

$$Cap(P_i) = (C_i, M_i, D_i, B_i). \tag{7}$$

Similarly, the resource capacity of a VM would be the resources allocated to it

$$Cap(V_j) = (C_j, M_j, D_j, B_j). \tag{8}$$

The additional resource requirement of a VM, V_j, would be (positive for expansion and negative for shrinking)

$$Require(V_j) = (C'_j, M'_j, D'_j, B'_j). \tag{9}$$

VM placement activity is performed while the VMs arrive in the cloud. Assume that the maximum number of VMs that a hypervisor can support is r. Out of the whole set of VMs V_j, only a few VMs in the subset V_s, s=1, 2,.....r, can be placed on a server P_i, if

$$Cap(P_i) \geq \sum_{s=1}^{r} Cap(V_s). \tag{10}$$

Moreover, all of the following should also hold:

$$C_i \geq \sum_{s=1}^{r} C_s \tag{11}$$

$$M_i \geq \sum_{s=1}^{r} M_s \tag{12}$$

$$D_i \geq \sum_{s=1}^{r} D_s \tag{13}$$

$$B_i \geq \sum_{s=1}^{r} B_s. \tag{14}$$

The requirement of VMs is met by idle resources. If the subset V_s is successfully placed on P_i, then the idle resources on P_i would be

$$Idle(P_i) = Cap(P_i) - \sum_{s=1}^{r} Cap(V_s). \tag{15}$$

Moreover, once the resources are allotted to VMs, during their runs, they will be continuously monitored using an auto-scaling algorithm. Additional resource requirement (Eq. (9)) will be considered for the fulfillment (out of idle resources calculated in Eq. (15)) if the following holds true:

$$Idle(P_i) \geq \sum_{s=1}^{r} Require(V_s). \tag{16}$$

Similarly, the idle resources of a VM should be removed to help the economic viability of cloud solutions. This removal will add the idle resources to the idle pool of resources of the cloud to further allot them to needy consumers when needed. An “overload” state would arise if one or more equations out of Eqs. (11)–(14) do not hold true for one or more VMs. Fig. 3 shows a detailed depiction of resource scaling and shrinking in a cloud infrastructure. Eq. (17) shows how the overload, underload, and normal load states affect $Cap(V_s)$ using the auto-scaling strategy. U is a utilization metric usually based on CPU usage; however, it may be a different metric based on one or more similar matrices. Identifying “overload” or “underload” conditions is usually decided by the user requirements and by the static and dynamic thresholds such as CPU utilization, response time of the web application, and file upload time [29]. V_{add} and V_{remove} are dependent upon the requirement and supported scaling techniques.

$$Cap(V_s) = \begin{cases} Cap(V_s) + Cap(V_{add}) & \text{if } U >= U_{overload}. \\ Cap(V_s) - Cap(V_{remove}) & \text{if } U <= U_{underload}. \\ Cap(V_s) & \text{if } U = U_{normalload}. \end{cases} \tag{17}$$

Auto-scaling would perform one of the following three possibilities or a combination of them.

1. **Vertical scaling:** In this scaling method, resources are added on top of the VM at the same physical server. These resources are added from the $Idle(P_i)$ pool of resources on server P_i. In the presence of an attack, Eq. (16) may not hold true after regular vertical scaling. In that case, either horizontal scaling or migration is the only available option to respond to the demand.

Table 1
DDoS in the cloud: stakeholders.

Stakeholders	Attack target
Victim server	Direct
Victim server owner	Direct
Users of victim VMs	Direct
Co-hosted VMs	Collateral
Host physical server	Collateral
Other physical server(s) in the cloud	Collateral
VMs on other hosts	Collateral
Network and network devices	Collateral/direct
Users of co-hosted VMs	Collateral
Cloud provider	Collateral/direct
Cloud customers	Collateral

2. **Horizontal scaling:** In this strategy, usually a cloned instance of the VM is created on a server other than P_i. This scaling strategy is only applicable to multi-instance applications with load balancing scheme in place. Most of the cloud providers use horizontal scaling by quickly cloning the VM instances. The newly created instance will have standard resources from the pool of instances that the cloud provider supports. Finding out a candidate physical server $P_{candidate}$ to start a cloned VM instance is part of the VM placement problem. A large number of approaches [34] are available to find a server that supports the required resources for a new VM instance by justifying the requirements with the idle resources on the server. A minimum resource requirement of a candidate server has the required idle resources to support the VM instance (Eq. (18)).

$$Idle(P_{candidate}) \geq \sum_{s=1}^{r} Require(V_s). \tag{18}$$

3. **Migration to another server:** This method is quite similar to horizontal scaling except that it does not require the creation of an instance. The same running instance is migrated to another server where the required resources are available (Eq. (18)). Therefore, the problem of identifying a candidate server remains the same as in the case of horizontal scaling. There are multiple factors that need to be considered before performing a migration. Some of these factors include the possible future requirements of resources, consequent migrations or swaps, and the migration costs.

4. Motivation and planning of the characterization

In this section, we present the motivation and planning of the experiments to characterize and quantify the effects. The major aim of these experiments was to see whether the non-target stakeholders of a cloud infrastructure are affected by a DDoS attack. Additionally, the aim of these experiments was to quantify and assess the damage caused. We have identified all the important actors in the DDoS attack scenario in the cloud (also shown in Fig. 1). These important components are listed in Table 1.

The impact characterization experiments were planned in such a manner that the following four categories of effects could be observed in detail: performance issues, additional costs incurred, indirect effects, and effects that are invisible at the moment but have a long-term impact.

1. **Performance**: The performance of a hosted service in a cloud may have multiple factors to measure performance or service quality. It may range from response time to number of concurrent users, timeouts, failures, and number of sessions. Accessibility and timely response are two of the most important factors for a web service. These factors are based on other factors including the design of the web page, server performance, and network bandwidth. Conversion rate and web reputation are generally associated with the performance of web service response time [35,36]. The major aim of our experiments was to determine the performance penalties on all components of a cloud platform while an attack was present. Factors that affect performance include arrival rate of incoming requests, fewer resources, and other performance-deteriorating functions like migration and swaps.
2. **Costs**: Cloud computing infrastructure is one of the most sought-after technology platforms for enterprises today. This is mostly due to the cost benefits it provides to VM owners and the resultant return on investment (ROI) [37]. In "pay-as-you-go" pricing models, costs are directly associated with resources used. CPU, memory, storage, and bandwidth are four important resources that are used in the cost calculation. This is an important factor to look for when considering the economic aspects of EDoS.
3. **Collateral/indirect effects**: These are the effects that are to be quantified on the non-targets, which are components of cloud architectures except the victim VM. The effects of the interest include both performance and cost-based effects.
4. **Invisible effects**: These are effects that are not visible in the first instance when the attack has appeared. End-user satisfaction, quality of service, downtime impact on business and long-term impact on reputation, penalties, and disputes are a few of the important impacts that are not visible directly while the attack is occurring. In most of the instances, these effects are not measurable as their real impact is dependent upon multiple factors including business agreements and time.

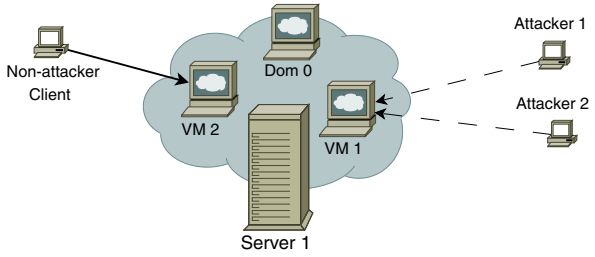


Fig. 4. Experiment setup 1: single physical server.

5. Experiments and results

Effect quantification is dependent upon multiple factors such as the size of the cloud, applications, resource allocation strategies, type of attack, and attack strength and its duration. For an effective understanding and classification, the following two experiment sets were planned. The first experiment was planned on a single physical server hosting multiple VMs to quantify server level local effects. The second experiment set was a cloud-scale experiment that was conducted to see the effects of DDoS/EDoS on cloud infrastructure as a whole. This would allow us to see the attack effects from both a local and an abstract perspective.

5.1. Experiment set 1: single physical server

The main aim of this experiment was to study the attack impact on a DDoSed victim server, co-hosted VMs, and the physical server hosting these VMs. This experiment was conducted using the setup shown in Fig. 4 and the configuration given in Table 2. For this, a web application's performance was compared when it was hosted alone (Experiment 1A), hosted with another VM with a similar load (Experiment 1B), and hosted with another VM that was under attack (Experiment 1C). These three experiments

Table 2
Experiment setup 1: single physical server.

Item	Configuration
Physical server	HP EliteDesk i7 3 GHz
Total CPUs	(4 Cores, 8 VCPus)
Total memory	4GB
Hypervisor	XenServer 6.2
Host/Guest/Attacker OS	Ubuntu 14.04 Server
Host CPUs	4 VCPUs (any CPU)
Host memory	732MB
Guest configuration	As specified in Fig. 5
Guest application	Apache2
Attackers	Dual Core (2GB)
Attacker application	ApacheBench2
Request concurrency	1/10/50/250 concurrent
Total requests	1000 in each set
Request size	2MB
Network	100 Mbps
CPU Affinity	Pinned as specified in Fig. 5

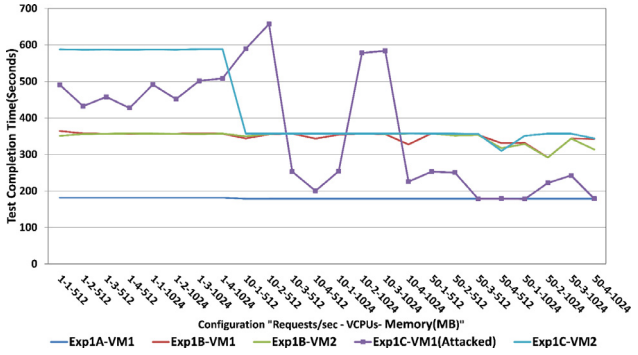


Fig. 5. Experiment 1: comparison of test completion times.

would enable us to understand the performance variations after the application's co-hosted VM started suffering from a DDoS attack. A test would comprise 1000 requests to be sent by clients to the web server. These requests would be sent to the web server with different concurrent requests (1/10/50 requests/s representing low/moderate/high load, respectively).

The attack consisted of 2000 requests (1000 requests each from two attackers). Each attacker was sending these requests with a concurrency of 250 requests, each of which generated a very heavy load for the given resources. The attacker traffic was planned using the guidelines given in the popular literature [38].

5.2. Results: single physical server

The completion time of each test is shown in Fig. 5 and Table 5 for various resource combinations across the three tests. For a web server serving static pages, the basic resources are the bandwidth and disk reads/transfers. Therefore, there is not much impact of increasing CPUs and memory. In the case of a dynamic web server, the basic resources may be extended to CPU cores and memory. Results showed that there was a significant increase in the completion time of the test due to contention. The resource contention race became tough from Experiment 1A to Experiment 1B and even tougher in Experiment 1C. Similarly, a significant rise can be seen in Table 4 for data transfer rate, connection times, and requests/second. At a later stage, the test completion time decreased owing to request failures. Therefore, victim VM1 could not respond to many of the requests (failures listed in Fig. 3). This shows the visible effects on the victim server and co-hosted VMs. Other than performance interference by a DDoSed VM, the behavior would trigger auto-scaling (based on the increased response time and requests), resulting in economic losses to the co-hosted VMs even though they were not directly DDoSed.

5.3. Experiment 2: cloud scale

To characterize the overall impact of DDoS attacks in an infrastructure cloud, we conducted a comprehensive cloud-scale experiment. The setup is as shown in Table 5 and is similar to that shown in Fig. 1. The major aim of this experiment was to study the effects on the cloud after some or more DDoSed VMs were introduced in the cloud. The authors of CloudSim [34] developed multiple schemes related to host overloading detection, VM selection for migration, host underloading detection, and VM placement, and tested them with PlanetLab and random traces. As shown in Table 5, there are five overload detection algorithms and four VM selection algorithms for migration implemented by Beloglazov and Buyya [34], making 20 combinations in all by choosing these algorithms (5*4). These combinations are governed by a few constants and input parameters, which were attached to their names by Beloglazov and Buyya [34], like iqr-mc-1.5. While aiming to effectively show the effects, we realized that this would be an ideal setup to use as it can help evaluate the effects of DDoS/EDoS on all these strategies to get a comprehensive insight. To conduct real quantification and evaluation, we used PlanetLab traces comprising the CPU utilization of VMs with an interval of 5 min. Three sets (three different days) were chosen in these experiments, indicating low (898 VMs), moderate (1052 VMs), and high (1516 VMs) load test cases. The average utilization in all of the PlanetLab traces was below 50%. The DDoSed VMs were introduced by inserting VMs with heavy utilization (100%). It is well established that CPU utilization of this order is achieved during DDoS attacks on computational resources [39]. Two most important metrics for evaluating the performance of a cloud, energy consumption and VM migrations, were evaluated by introducing a few or more DDoSed VMs.

5.4. Results: cloud scale

The results of three sets of VM traces are shown in Figs. 6–8. A few result items for set 1 (1052 VMs) are given in Table 6 for reference to the charts. Both energy consumption and number of VM migrations were plotted with different numbers/shares of DDoSed VMs among the normal VMs and 20 sets of overload detection and VM selection algorithms. The non-power-aware scheduler showed 2410 kWh of consumption for all combinations, and DVFS ranged between 600 and 1200 kWh from 0 to 100% of DDoSed VMs (not plotted in the charts). The energy consumption charts show an almost linear increase with the increase in the DDoSed VMs in all the strategies. Usually, energy consumption is a function of CPU utilization and that is evident in the charts. Similarly, service pricing/costs based on energy may also be calculated. The number of VM migrations was increasing till a certain point (5-10% of DDoSed VMs) and then it started decreasing and reached a minimum number of migrations. The reason for this increase was the introduction of DDoSed VMs at multiple servers in the cloud, followed by the overload situation of VMs given in Eqs. 11–15 requiring migrations (creation of new instances was not implemented in the simulation environment). After having 10–20% of attacked VMs, the number of overloaded VMs increased but the places or other candidate servers for hosting these VMs decreased. Mostly, the reason is that the other servers were also getting stressed and were not in a position to support the requirements of incoming migratory VMs (Eq. (16)). With an insertion of 5–10% of attacked VMs, the corresponding increase in the number of VM migrations was near 50% (23 K to 34 K). Cloud-scale DDoS attacks and their possibility cannot be predicted, at least for public clouds. There are multiple recent incidents related to Amazon, Greatfire.org, Linode, and

Table 3
Experiment 1: comparison of test completion times.

Configuration R: Requests V: vCPUs M: Memory (MB)			Experiment 1A	Experiment 1B: two VMs		Experiment 1C Two VMs and VM1 (attacked) F: Request failures		
R	V	M	T_{VM1}	T_{VM1}	T_{VM2}	T_{VM1}	F	T_{VM2}
1	1	512	181.9 s	364.3 s	351.2 s	491.0 s	0	587.7 s
1	2	512	181.9 s	357.8 s	355.8 s	432.5 s	90	587.0 s
1	3	512	181.8 s	356.5 s	356.7 s	457.8 s	0	587.3 s
1	4	512	181.8 s	357.0 s	357.6 s	427.4 s	357	586.8 s
1	1	1024	181.7 s	356.6 s	356.9 s	491.7 s	0	587.5 s
1	2	1024	181.6 s	356.6 s	356.7 s	451.9 s	0	587.0 s
1	3	1024	181.8 s	357.9 s	355.9 s	501.9 s	0	588.4 s
1	4	1024	181.8 s	356.9 s	357.3 s	508.5 s	2	588.3 s
10	1	512	178.7 s	343.9 s	349.8 s	589.5 s	468	357.3 s
10	2	512	178.8 s	355.3 s	356.2 s	657.4 s	327	357.1 s
10	3	512	178.8 s	357.3 s	357.3 s	252.7 s	0	357.3 s
10	4	512	178.8 s	343.3 s	357.5 s	200.5 s	96	357.3 s
10	1	1024	178.8 s	354.1 s	357.4 s	254.0 s	0	357.4 s
10	2	1024	178.7 s	357.3 s	357.4 s	578.1 s	864	357.1 s
10	3	1024	178.7 s	355.5 s	356.2 s	584.0 s	825	357.2 s
10	4	1024	178.7 s	327.7 s	357.5 s	225.5 s	54	357.6 s
50	1	512	178.9 s	357.7 s	357.5 s	252.6 s	381	357.3 s
50	2	512	178.8 s	351.8 s	351.8 s	250.4 s	396	357.0 s
50	3	512	178.9 s	353.4 s	353.4 s	178.8 s	0	356.3 s
50	4	512	178.9 s	331.3 s	317.4 s	179.0 s	18	309.7 s
50	1	1024	179.0 s	331.8 s	328.6 s	177.9 s	75	351.2 s
50	2	1024	178.7 s	291.8 s	291.6 s	221.9 s	387	357.3 s
50	3	1024	178.8 s	343.7 s	343.6 s	242.4 s	387	357.0 s
50	4	1024	178.8 s	343.2 s	313.4 s	178.9 s	15	344.0 s

Table 4
Comparison of performance matrices in experiment 1.

Experiment, VM	Transfer rate (Kbytes/s)	Served Requests/s	Connection times (ms)				
			Min	Mean	+/-SD	Median	Max
1A, VM1	11,260.32	5.5	180	182	2.4	181	223
1B, VM1	5832.58	2.85	181	351	34.6	357	431
1B, VM2	5621.92	2.74	181	364	42.7	357	602
1C, VM1	4171.77	2.04	10,811	107,510	38,643.3	107,256	389,555
1C, VM2	3485.17	1.7	564	588	13.1	585	726

Rackspace, which are discussed in Section 1. Additionally, multiple works such as [40–42] have strongly reported and evaluated the consequences of cloud-scale attacks.

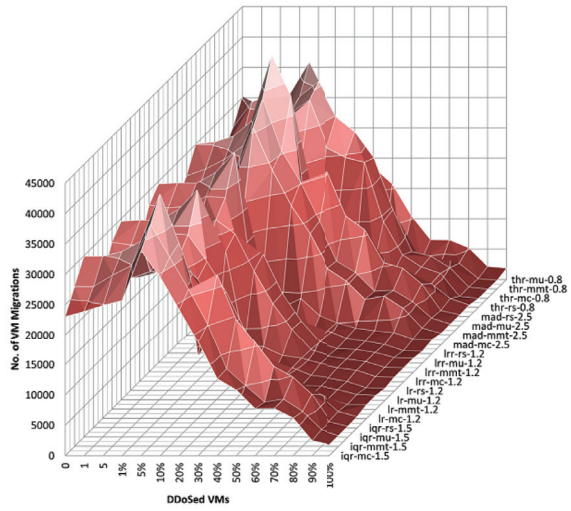
6. Collateral damages and contributors: discussion

The experiments in Section 5 have shown that the indirect effects on other components of the cloud must not be neglected. The traditional DDoS mitigation mechanisms would not help here as the cloud context would require control over resource provisioning and auto-scaling. The following is a summary of the effects that appeared during the experiments, which should be kept in mind while developing services and mitigation solutions in the cloud.

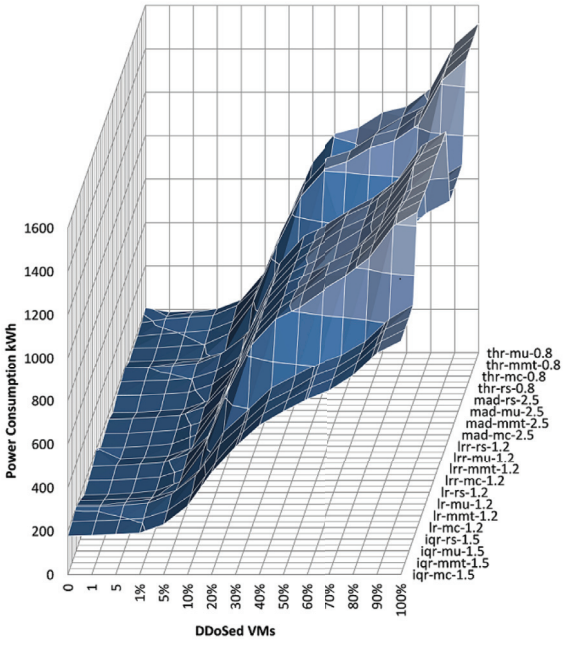
1. *Victim VM*: Economic losses that may ultimately reach service denial; unnecessary resource buying; performance issues such as low throughput, high response time, and request failures; service downtime; and short-term and long-term business and reputation losses.
2. *Co-hosted VMs*: Indirect EDoS, unnecessary resource buying and performance interference, unnecessary migration/VM instance creation, and performance issues such as low throughput, high response time, and request failures.
3. *Host physical server*: Resource overload situation and higher power consumption.

Table 5
Experiment setup 2: cloud scale.

Item	Configuration
Simulation environment	CloudSim 3.0.3
No. of servers	800
VM Traces	"PlanetLab" traces of 3 days
No. of VMs in the cloud	Set 1: 1052 VMs (03-03-2011) Set 2: 1516 VMs (22-03-2011) Set 3: 898 VMs (06-03-2011)
Server configurations	1. HP ProLiant ML110 G4 Intel Xeon 3040, 2 cores 1860 MHz, 4GB (400 servers) 2. HP ProLiant ML110 G5 Intel Xeon 3075, 2 cores 2660 MHz, 4GB (400 servers)
Overload detection algorithms	THR: Threshold MAD: Median absolute deviation IQR: Interquartile range LR: Local regression LRR: Robust local regression
VM Selection algorithms	MMT: Minimum migration time MC: Minimum correlation RS: Random selection MU: Minimum utilization
Other algorithms	DVFS: Dynamic voltage frequency scaling NPA: Non-power aware

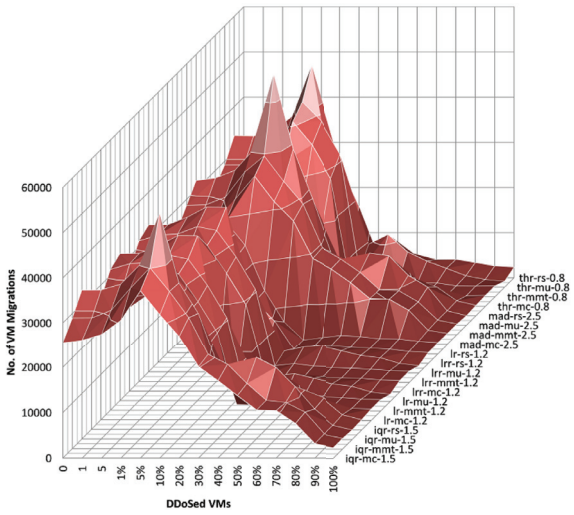


(a) VM migrations

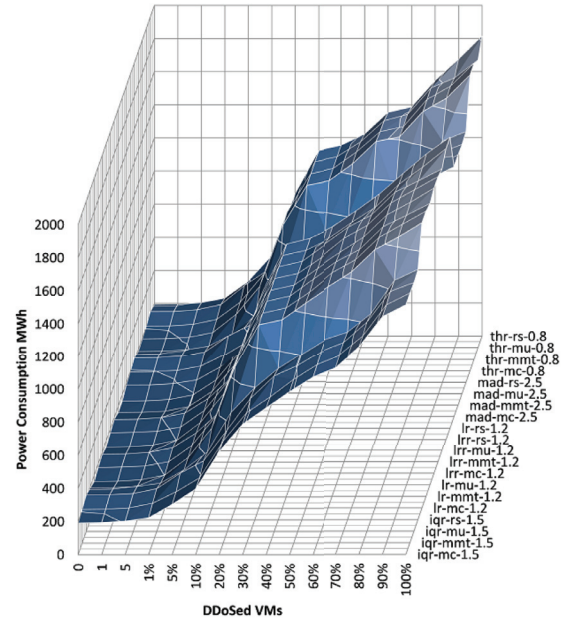


(b) Energy consumption

Fig. 6. Results of experiment 2: set 1 (1052 VMs).



(a) VM migrations



(b) Energy consumption

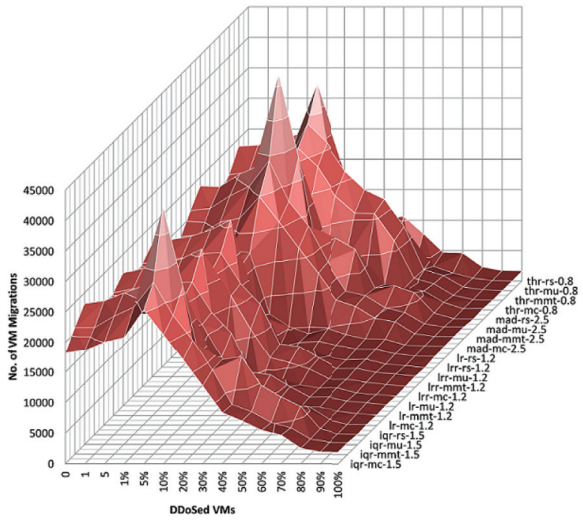
Fig. 7. Results of experiment 2: set 2 (1516 VMs).

4. *Other physical server(s) in the cloud*: Indirectly affected by migrated VMs and VM instance creation, which may result in overload.
5. *VMs on other hosts*: Indirectly affected by migrated and new VM instances that became co-hosts to these VMs.
6. *Cloud as a whole*: Heavy energy consumption, running costs, cooling costs, VM migrations, service-level agreement (SLA) violations, and business losses.
7. *Network and network devices*: Heavy bandwidth consumption, network losses, and poor quality of service.
8. *Users of victim VMs*: High response time, poor quality of service, service downtime, and related business/reputation impacts on dependent services.
9. *Users of co-hosted VMs*: High response time, poor quality of service, service downtime, and related business/reputation impacts on dependent services.
10. *Cloud customers*: Business losses, SLA violations, service health, and business difficulties; the cloud could have accommodated and run more VMs; requests to create additional VM instances for existing customers might not be fulfilled.

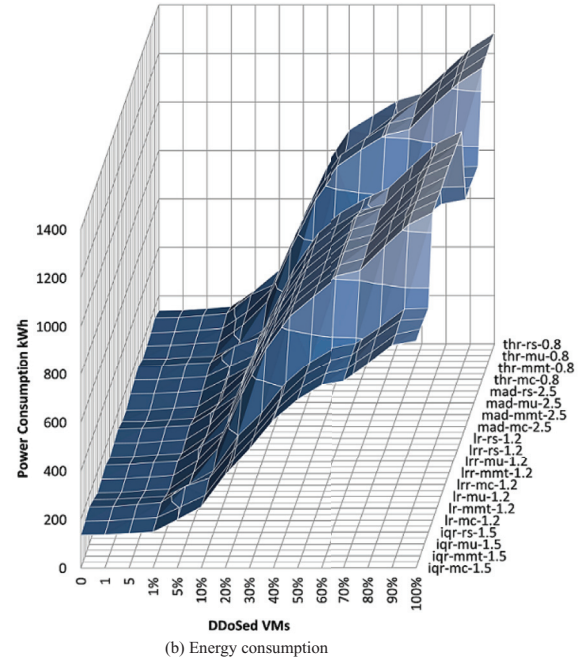
In the following section, we identify the specific contributors to the characterized collateral damages. These observed scenarios/effects are the major reasons for these effects.

6.1. Performance interference

Performance isolation is a property provided by virtualization [43,44]. Resource contention among VMs for the basic resources and the resulting performance contention are also important effects to be considered. Performance contention is an outcome of resource sharing. In the presence of a DDoSed VM, which may be mainly stressing a specific resource, say, a disk data transfer, co-hosted VMs would also experience difficulty regarding turnaround time for disk transfers. Web services are usually considered mixed load applications, and, hence, this contention may



(a) VM migrations



(b) Energy consumption

Fig. 8. Results of experiment 2: set 3 (898 VMs).

be visible for multiple resources such as the CPU, disk, and bandwidth.

6.2. Resource race among VMs

Let us take the case of server 1 in Fig. 1. This server is running four web server VMs that belong to four different organizations or owners taking advantage of the multi-tenant cloud. When VM1 is DDoSed by attackers, it will ask for more resources and these will be added using auto-scaling. If the attack continues, it will ask for more resources again and again, thus acquiring the maximum that is available on server 2 for a VM. Finally, VM1 will be flagged for either additional instance creation or migration at other servers (Fig. 3). At the same time, there may be a stage where other co-hosted VMs—VM2, VM3, and VM4—have some real requirement for resources because of real web requests and will ask for the same. Obviously, this requirement cannot be met by the same physical server and these VMs may also be flagged for either migration or additional instance creation on some other servers. Creating another instance or migrating to another server is an advanced memory transfer/copy process and incurs a downtime. Additionally, they incur resource usage, overhead costs, and downtime of the services. Although resource race is part of a multi-user, multi-tenant system, this fake race, however, appeared only because of one DDoSed VM on the server. The co-hosted VMs that are not a victim of the attack may also need to get migrated or start another instance because of this situation.

6.3. VM placement and load balancing

Most of the VM placement algorithms place VMs in a cloud using an optimization method such as dynamic bin packing [45]. Live VM migration is also used as one of the activities of auto-scaling (Fig. 3). As discussed in Section 6.2, DDoSed VMs may result in unnecessary migrations of one or more VMs. There are two possibilities: first, the DDoSed VM may be migrated as a direct effect of DDoS; second, other VMs may be migrated owing to the decision of the migration decision-making algorithms such as minimum migration time (MMT) and minimum utilization (MU) [34]. In the worst case, there might be a swap of VMs or multiple migrations/swaps (reshuffles) among physical servers for an effective load management and resource allocation. To summarize, the following detailed consequences may occur:

1. **New instance/clone creation**: In case the auto-scaling algorithm chooses to create a new instance for a DDoSed VM, the effect will be epidemic and will adversely affect many VMs and the network bandwidth. This can be seen in Fig. 9a, where the VM under attack (on server 1) is dynamically scaled by creating VM instances on server 2 and server 3. In case a new instance is created for other VMs, obviously, they will be charged for it, resulting in an “indirect DDoS/EDoS.”.
2. **Effect spread and migrations**: A DDoS VM, if migrated owing to the “overload” situation, and if the attack continues, will be migrated again soon to some other server and will continue getting migrated. This will spread the additional effects on other co-hosted VMs and servers epidemically. This scenario is shown in Fig. 9b, where the VM under target is getting migrated from Server 1 to Server 2 and again to Server 3.
3. Migrant selection: It may happen that VMs, except the victim VM, may be selected for migration to other servers. This will directly affect the services offered by them owing to migration cost, downtime, and related overheads. Moreover, if they are migrated because of fake alarms such as a higher response time for a web server, these VMs will suffer from economic losses indirectly, as they are buying extra resources. This extra buying is due to the shared resource being stressed by the co-hosted DDoSed VM.
4. Migrations, swaps, and shuffles: If migrations are converted into VM swaps, it will be a fatal and exhaustive effect for multiple physical servers and VMs. VM swaps are a reality of cloud environments and are needed if a migration is not able to cope with the requirement. A load balancing heuristic is needed to balance the load between two or more physical servers by migrating VMs in between [46]. Large performance penalties will appear on the network owing to the migrations/swaps.

7. Solution space

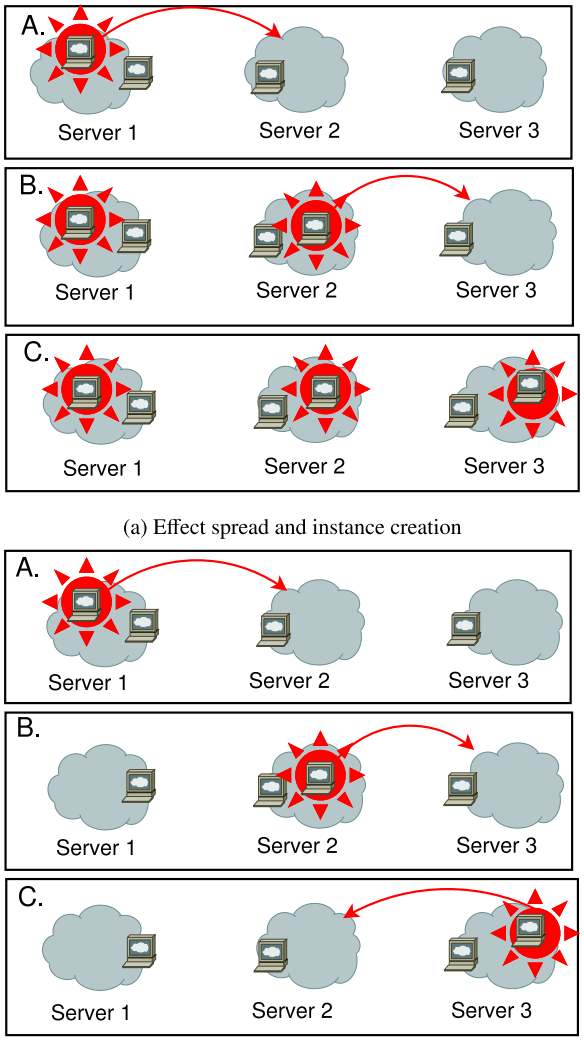
In this section, we give a detailed guideline and direction for solution design. There are a few recent contributions that meet one or a few of the defined requirements. For a detailed survey of DDoS solutions and requirements in cloud computing, see [47]. We refer to them in each of the requirement in which they have made a contribution. However, there is an immense need for solution development that takes all these aims into consideration. We

Table 6
Experiment 2: DDoS effects on set 1 of 1052 VMs.

	Energy consumption (kWh)							
Algorithms	0	1	5%	10%	20%	30%	50%	100%
DVFS	803.91	788.92	794.81	870.54	906.79	1000.49	1067.91	1205.67
iqr-mc-1.5	177.1	179.9	229.69	314.8	467.58	586.54	749.15	1077.42
mad-mc-2.5	176.13	177.45	227.01	311.05	461.81	583.35	747.78	1077.42
thr-mu-0.8	206.73	186.33	243.88	373.27	631.33	878.07	1037.31	1515.85
	No. of VM migrations							
Algorithms	**0**	**1**	**5%**	**10%**	**20%**	**30%**	**50%**	**100%**
DVFS	0	0	0	0	0	0	0	0
iqr-mc-1.5	23,035	23,881	33,924	29,542	24,142	18,717	10,796	1784
mad-mc-2.5	23,691	24,452	35,828	28,801	22,985	18,343	12,534	1784
thr-mmt-0.8	26,634	30,825	37,323	28,113	4119	5133	5267	1392

draw lessons from the effect characterization studies given in this work to design this solution space. The following are the major requirements of an efficient DDoS mitigation solution in the cloud, which also aims at minimizing the indirect effects of DDoS attacks on other stakeholders.

1. **Strong isolation**: Performance isolation and resource isolation are two areas that require a thorough relook and design consideration to achieve a strong isolation. To achieve this, we need fair resource sharing and accounting. Pinned or dedicated resources are also one important alternative to provide isolation; however, it is a costly solution that is limited to a few resources such as CPUs and bandwidth [43,44]. Similar requirements for a strong isolation are provided in [48].
2. **Victim separation**: Cloud incident management systems should handle both incoming DDoS to a VM and any malicious internal VM sending DDoS traffic to an outside network. We need mechanisms at multiple levels to identify the victim and separate it from the other tenants to minimize the effects. Reserved resources, shutdown, and backup servers are a few available solutions in this direction. The solutions provided by the authors in [10,49,50] are some of the important contributions that advocate additional resources for mitigation. The authors in [51] provide a victim separation method that migrates the server to backup resources and isolates the service. Additionally, the server is migrated back to its original resources once the attack is over. This approach is similar to shutting the server down, where no efforts are spent to stop the attack. The authors in [52] also provide a backup resource based on a low-cost solution to use resources over the untrusted cloud.
3. **DDoS-aware auto-scaling**: We can see in both the experiments and the system model that the major cause of EDoS attack is incorrect decisions made by the auto-scaling algorithm. Auto-scaling algorithms must be designed by keeping DDoS utilization surges in mind. It has been quite a difficult problem to differentiate DDoS traffic from legitimate traffic. However, it is comparably easier to decide the presence of an attack. This fact may help in devising EDoS-aware auto-scaling algorithms [53]. The authors in [53] have provided a solution that uses the presence of the DDoS attack and subsequently decides whether the requirement is due to legitimate traffic or to attack traffic. This helps in wisely taking auto-scaling decisions, while at the same time serving legitimate customers.
4. **Collaborative solutions**: Internet service provider, cloud/hypervisor, network, VM, and application are the five layers of the solution space that, if done collaboratively, can detect and mitigate the attack effects with high assurance. Multi-level solutions with one or more of these levels have been tried by the solutions in [10,49,54]; however, efforts are needed to minimize the additional effects proposed in this work.



(a) Effect spread and instance creation

(b) Effect spread and migrations

Fig. 9. Contributors to collateral damages (starred VM is the victim VM).

5. **Verifiable and fine-grained accounting**: Issues related to loss sharing and dispute resolution relate to proper accounting. Most of the cloud players in the market have fixed pricing models that are based on hourly metering. There is an immense need to have fine-grained and verifiable (at the VM end) accounting methods that show the real resource usage. This is especially needed for the resources that are shared. This will help in implementing the pricing models that follow the

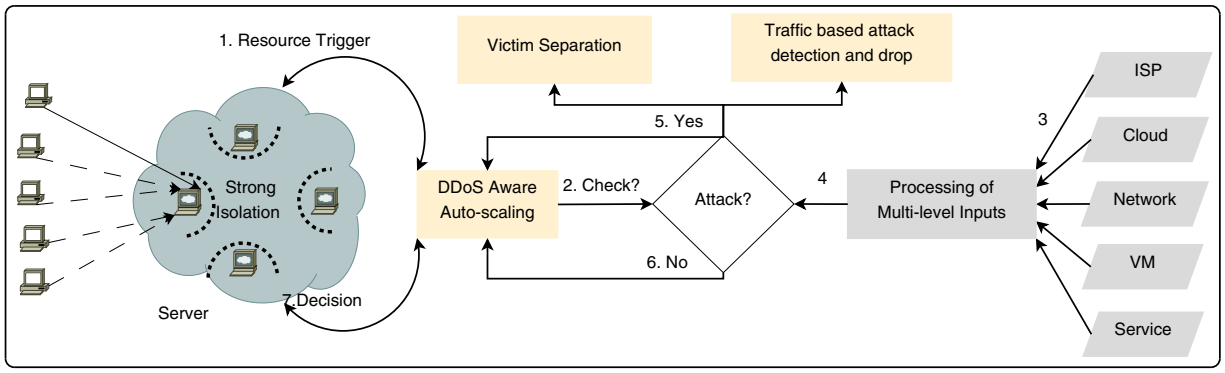


Fig. 10. Solution space: DDoS attack mitigation and minimization of collateral damages.

"pay-as-you-go" billing models. A few important solutions in these directions are given in [55–57].

We have provided an example solution that has the required ingredients to address these requirements in Fig. 10. We have made an effort to incorporate all the requirements given in Section 7 in this design. One important observation is that the past contributions in the literature mostly target traffic differentiation to identify the attackers and block them. On the other hand, we argue that, to overcome the collateral damages, we need to have other important features in addition to traffic differentiation. The primary requirement of our solution is a strong isolation among the co-hosted VMs. This requires no resource contention and performance interference.

1. A resource increase trigger will invoke the auto-scaling algorithm while the attack is in place.
2. Now the auto-scaling algorithm, which is DDoS aware, would ask the attack decision module to check the presence of the attack.
3. The attack decision module has a multilevel information and mitigation system in place, which may help in providing solutions at all the ends.
4. This information is processed and sent to the attack decision module to take a decision.
5. Once it has been detected that there is an attack, the decision module may want to trigger a few important processes.
   (A) There is no resource increase to serve the attack traffic. However, there might be a resource increase required to serve the new benign traffic or fasten the mitigation process.
   (B) The victim should be separated with a dedicated resource plan, in which resources are not shared with any other VMs.
   (C) Traditional traffic-based filtering can be applied to do low-level mitigation.
6. If there is no attack, then the auto-scaling scheme may take decisions according to its traditional auto-scaling practices to maintain service availability.

As our study is a characterization and effect orchestration study, we leave the evaluation and analysis of the solution space open for future research.

8. Related works

Most of the works that have contributed in the area of "DDoS" attacks in the cloud have tackled these attacks similar to how they are treated in fixed on-premise infrastructures. The impact of virtualization, multi-tenancy, on-demand resources, and pricing is a factor that changes the whole scenario. There are a large number of classical surveys available to study DDoS and its mitigation methods [24,58,59]. Works toward the impact and effect characterization of DDoS attacks in cloud computing are listed here. The authors in [54] have demonstrated through experiments that network DDoS attack may affect the power usage of a server by considering the CPU and I/O usage. The authors have evaluated the attack on a server with varied intensity of the attack. A related but extended work has been shown by [60] for I/O-based DDoS attacks. Shea and Liu in [61] reported the impact of DDoS attack on various virtualization strategies such as paravirtualization, full virtualization, and hardware VM, and identified serious performance vulnerabilities as compared to traditional infrastructure.

As an extension to their work, Shea and Liu [62] quantified the performance degradation of a cloud-based virtualized web server and compared it with a fixed infrastructure web server. They identified that a VM-based server has large performance penalties (due to virtualization and related overheads) during a DDoS attack. The authors in [63] showed the performance penalties for multi-tenant virtualized networks. Security issues due to multi-tenancy were discussed in [64] to provide security to a VM from co-hosted VMs. H. Liu in [40] identified a new form of attack toward cloud data centers where the attacker's main motive is to choke the cloud bandwidth. This work has shown the initial directions of cloud-scale attacks. Yu et al. in [10] proposed a cloud DDoS mitigation solution based on the idle resources in the cloud. Although this solution is an early work toward mitigating a DDoS attack on a cloud consumer and works toward a resource allocation algorithm, like other works, however, this does not consider the effects on other cloud components. In another work, Yu et al. [65] proposed an economic cloud firewall that offers a trade-off between cloud resources and the required quality of service. A different perspective in this direction was contributed by the authors in [42], which showed cases where problems in SNMP-managed data center cooling systems had resulted in fatal effects from a DDoS attack. The authors showed through simulations that DDoS attacks during inactivity of proper cooling systems can result in a disastrous situation termed as "data center meltdown." Similar to this, the authors in [41] identified the possibilities of power-based denial/power outage attacks. They conducted simulations to show the possibilities of these attacks on cloud data centers and provided solutions. The authors in [66] evaluated the impact of EDoS on an instance providing cloud-based services and showed through simulations that cost considerations are quite heavy while a server is under EDoS. Miao et al. proposed DDoS mitigation systems implemented over the Nimbus architecture [67].

An interesting experiment was conducted by the authors in [68] to determine the maximum charges by applying an EDoS attack. The authors targeted a cloud service on Amazon Cloudfront and sent 1000 requests/s with 1000 Mbps for 1 month. It was an expensive attack for the service as the economic loss due to

this attack resulted in a bill of $42K. Palmieri et al. in [69] provided a detailed characterization of DDoS attacks in cloud computing from the perspective of fraudulent energy consumption. The authors showed an attack algorithm to orchestrate DDoS attacks that attempt to harm in terms of energy consumption. The authors also presented cases of sophisticated attack sequences that are intelligent enough to remain undetected by varying the attack properties. Ficco and Palmieri in [70] showed a method that exploits application-level weaknesses to organize a successful energy-oriented DDoS. Palmieri et al. also showed different directions related to these variants of DDoS attacks even without breaking the security firewalls [11]. These effects resulted in a heavy burden on firewalls. The authors in [12] showed a detailed stealthy attack mechanism and its effects. In this work, the authors showed through experiments that a slow and rising attack frequency and strength can become fatal with heavy financial costs owing to the flexibilities provided by the cloud.

Idziorek et al. [71] also showed a similar consequence of EDoS attack on cloud infrastructures. The authors tested both very low rate attacks and heavy attacks to test and characterize the actual effects. It was highlighted in the work that even a single request per minute by a single attacker can result in a total of 13 GB of data transfer, considering an average request response size of 320 KB. The authors treated the EDoS attacks in the cloud as fraudulent resource consumption attacks. The authors in [72] tested a virtualized web server running on Amazon EC2 instance and showed that economic losses are quite visible. The easy detection of botnets using blacklist databases has resulted in attacker's shift toward designing novel attack methods. Vlajic et al. [73] showed the planning of an EDoS attack using malicious web browsers across machines. This was achieved using popular techniques such as phishing and social engineering-based web bugs. These experiments have been shown on the Amazon S3 infrastructure. Similar studies have been conducted by other authors [66,74].

Most of the works related to DDoS characterization are limited to the effects on a victim VM or a cloud and their deteriorating performance. They do not consider the effects on non-targets as collateral damages. Performance penalties due to virtualization have not been considered in the perspective of DDoS/EDoS attacks. This work is the first one to propose a systematic analysis and effect characterization in this direction. Many providers have proposed initial solutions to DDoS/EDoS attacks by keeping environmental variables such as fixing maximum resource caps, fixed resources to VMs, and the maximum number of instances allowed. Certainly, these fixes cannot resolve these issues as they would behave in a manner similar to traditional fixed infrastructures, leaving the cloud features aside, and may also result in an early DDoS due to the fixed resources.

9. Conclusions and future scope

This work provides a novel insight into the effects of DDoS attacks in cloud computing. In addition to the obvious targets, which are either a victim server or a network, we have argued and shown that almost all the components and stakeholders of a cloud architecture are affected by a DDoS attack. Attack quantification depends on many factors, including the strength of the DDoS attack, victim application, and resources. We have developed a system model of cloud computing resource allocation to help in understanding the role of auto-scaling algorithms in a DDoS attack and its success. This model has also detailed the resource "overload" state of a VM under a DDoS attack and its possible spread using vertical scaling, horizontal scaling, and migrations. Furthermore, features such as auto-scaling, migration, multi-tenancy, resource race, performance interference, and isolation have been identified. These features multiply the impact of DDoS in virtualized infrastructure clouds. It has been shown that multiple unrelated and non-targeted VMs, servers, and users are also affected by a DDoS or EDoS attack in the cloud.

An effort has also been made to differentiate and correlate DDoS and its economic version, EDoS, with the help of factors such as the time to reach DoS and attacker targets. Attack effect spread, migrant selection and overhead of cloning, and migration and swaps were discussed to quantify these effects on experiments and their planning. We kept performance issues, costs, overhead, and invisible effects, and their assessment as major objectives of the experiments. To understand the effects from a microscopic view, we conducted a single-server experiment to determine the effects on the victim server and the co-hosted VMs. Additionally, to have a detailed abstract view, we performed cloud-scale simulations to determine the consequences of DDoS attacks using real workload traces of PlanetLab. Various algorithms of VM placement, VM migration, and resource allocation were configured with various shares of DDoSed VMs in the workload. These experiments have given an insight into the multiple indirect impacts of DDoS attacks on non-targets such as co-hosted VMs, host physical server, neighboring physical servers, cloud as a whole, network and network devices, and end users of all the services. Performance, cost, power, and various overheads were the major effects that were determined from these experiments. This study provides a strong motivation toward re-examining the design of cloud resource allocation algorithms and strong performance isolation. Additionally, systematic and specific efforts are needed in the direction of mitigating EDoS and DDoS attacks in cloud platforms. In the end, we provide a solution space and guidelines by creating a line of requirements for an efficient solution. We kept collateral damages in mind while designing these requirements. We assure the suitability of the solution space with the help of individual contributions made in the past. We keep the real evaluation of this design open to future contributions from the security research community. Smokescreening, malware spread, dispute resolution, IT chargeback, SLA designs, and loss sharing are some of the most important aspects that are completely open while looking at DDoS attacks in cloud computing.

References

[1] K. Labs, Global it security risks survey 2014 Distributed denial of service (DDOS) attacks, 2014, (http://media.kaspersky.com/en/B2B-International-2014-Survey-DDoS-Summary-Report.pdf).
[2] C. Burt, Large volume DDos attacks see exceptional growth in first half of 2014: arbor networks, 2014, (http://www.thewhir.com/web-hosting-news/large-volume-ddos-attacks-see-exceptional-growth-first-half-2014-arbor-networks).
[3] P. Nelson, Cybercriminals moving into cloud big time, report says, 2015, (http://www.networkworld.com/article/2900125/malware-cybercrime/criminals-moving-into-cloud-big-time-says-report.html).
[4] T. Seals, Q1 2015 DDos attacks spike, targeting cloud, 2015, (http://www.infosecurity-magazine.com/news/q1-2015-ddos-attacks-spike/).
[5] T. Robinson, Series of DDos attacks plague linode data centers, infrastructure, 2015, (http://www.scmagazine.com/).
[6] S. News, Survey - with DDos attacks companies lose around £100k/hr, 2015, (http://www.spamfighter.com/News-19554-Survey-With-DDoS-Attacks-Companies-Lose-around-100kHr.htm).
[7] L. Munson, Greatfire.org faces daily $30,000 bill from DDos attack, 2015, (https://nakedsecurity.sophos.com/2015/03/20/greatfire-org-faces-daily-30000-bill-from-ddos-attack/).
[8] R. Cohen, Cloud attack: Economic denial of sustainability (EDos), 2009, (http://www.elasticvapor.com/2009/01/cloud-attack-economic-denial-of.html).
[9] J. Idziorek, M. Tannian, D. Jacobson, Detecting fraudulent use of cloud resources, in: Proceedings of the 3rd ACM Workshop on Cloud Computing Security, ACM, 2011, pp. 61–72.
[10] S. Yu, Y. Tian, S. Guo, D.O. Wu, Can we beat ddos attacks in clouds? Parallel Distrib. Syst. IEEE Trans. 25 (9) (2014) 2245–2254.
[11] F. Palmieri, S. Ricciardi, U. Fiore, M. Ficco, A. Castiglione, Energy-oriented denial of service attacks: an emerging menace for large cloud infrastructures, J. Supercomput. 71 (5) (2014) 1620–1641, doi:10.1007/s11227-014-1242-6.
[12] M. Ficco, M. Rak, Stealthy denial of service strategy in cloud computing, Cloud Comput. IEEE Trans. 3 (1) (2015) 80–94, doi:10.1109/TCC.2014.2325045.
[13] W. Alosaimi, K. Al-Begain, An enhanced economical denial of sustainability mitigation system for the cloud, in: NGMAST, IEEE, 2013, pp. 19–25.

[14] W. Alosaimi, K. Al-Begain, A new method to mitigate the impacts of the economical denial of sustainability attacks against the cloud, in: Proceedings of the 14th Annual Post Graduates Symposium on the convergence of Telecommunication, Networking and Broadcasting (PGNet), 2013, pp. 116–121.

[15] S.H. Khor, A. Nakao, spow: On-demand cloud-based EDDos mitigation mechanism, in: HotDep (Fifth Workshop on Hot Topics in System Dependability), 2009.

[16] M. Masood, Z. Anwar, S.A. Raza, M.A. Hur, Edos armor: A cost effective economic denial of sustainability attack mitigation framework for e-commerce applications in cloud environments, in: Multi Topic Conference (INMIC), 2013 16th International, 2013, pp. 37–42, doi:10.1109/INMIC.2013.6731321.

[17] S. Ranjan, R. Swaminathan, M. Uysal, A. Nucci, E. Knightly, Ddos-shield: Ddos-resilient scheduling to counter application layer attacks, IEEE/ACM Trans. Netw. 17 (1) (2009) 26–39, doi:10.1109/TNET.2008.926503.

[18] M.H. Sqalli, F. Al-Haidari, K. Salah, EDoS-shield - A two-steps mitigation technique against EDoS attacks in cloud computing, in: UCC, IEEE Computer Society, 2011, pp. 49–56.

[19] F. Al-Haidari, M.H. Sqalli, K. Salah, Enhanced EDoS-shield for mitigating EDoS attacks originating from spoofed IP addresses, in: G. Min, Y. Wu, L.C. Liu, X. Jin, S.A. Jarvis, A.Y. Al-Dubai (Eds.), 11th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, TrustCom 2012, Liverpool, United Kingdom, June 25-27, 2012, IEEE Computer Society, 2012, pp. 1167–1174.

[20] V. Huang, R. Huang, M. Chiang, A DDos mitigation system with multi-stage detection and text-based turing testing in cloud computing, in: Advanced Information Networking and Applications Workshops (WAINA), 2013 27th International Conference on, IEEE, 2013, pp. 655–662.

[21] T. Karnwal, T. Sivakumar, G. Aghila, A comber approach to protect cloud computing against XML DDos and HTTP DDos attack, in: Electrical, Electronics and Computer Science (SCEECS), 2012 IEEE Students' Conference on, IEEE, 2012, pp. 1–5.

[22] M.N. Kumar, P. Sujatha, V. Kalva, R. Nagori, A.K. Katukojwala, M. Kumar, Mitigating economic denial of sustainability (EDos) in cloud computing using in-cloud scrubber service, in: Proceedings of the 2012 Fourth International Conference on Computational Intelligence and Communication Networks, in: CICN '12, IEEE Computer Society, Washington, DC, USA, 2012, pp. 535–539, doi:10.1109/CICN.2012.149.

[23] L. Feinstein, D. Schnackenberg, R. Balupari, D. Kindred, Statistical approaches to DDos attack detection and response, in: DARPA Information Survivability Conference and Exposition, 2003. Proceedings, vol. 1, IEEE, 2003, pp. 303–314.

[24] J. Mirkovic, P. Reiher, A taxonomy of DDos attack and DDos defense mechanisms, SIGCOMM Comput. Commun. Rev. 34 (2) (2004) 39–53, doi:10.1145/997150.997156.

[25] J. Mirković, G. Prier, P. Reiher, Attacking DDos at the source, in: Network Protocols, 2002. Proceedings. 10th IEEE International Conference on, 2002, pp. 312–321, doi:10.1109/ICNP.2002.1181418.

[26] T. Peng, C. Leckie, K. Ramamohanarao, Survey of network-based defense mechanisms countering the DoS and DDos problems, ACM Comput. Surv. 39 (1) (2007), doi:10.1145/1216370.1216373.

[27] H. Wang, K.G. Shin, Transport-aware IP routers: a built-in protection mechanism to counter DDos attacks, Parallel Distrib. Syst. IEEE Trans. 14 (9) (2003) 873–884, doi:10.1109/TPDS.2003.1233710.

[28] G. Somani, M.S. Gaur, D. Sanghi, Ddos/edos attack in cloud: Affecting everyone out there!, in: Proceedings of the 8th International Conference on Security of Information and Networks, in: SIN '15, ACM, New York, NY, USA, 2015, pp. 169–176, doi:10.1145/2799979.2800005.

[29] M. Stillwell, D. Schanzenbach, F. Vivien, H. Casanova, Resource allocation algorithms for virtualized service hosting platforms, J. Parallel Distrib. Comp. 70 (9) (2010) 962–974.

[30] L.M. Vaquero, L. Rodero-Merino, R. Buyya, Dynamically scaling applications in the cloud, SIGCOMM Comp. Commun. Rev. 41 (1) (2011) 45–52.

[31] F. Al-Haidari, M. Sqalli, K. Salah, Impact of CPU utilization thresholds and scaling size on autoscaling cloud resources, in: Cloud Computing Technology and Science (CloudCom), 2013 IEEE 5th International Conference on, vol. 2, IEEE, 2013, pp. 256–261.

[32] M. Mao, J. Li, M. Humphrey, Cloud auto-scaling with deadline and budget constraints, in: Grid Computing (GRID), 2010 11th IEEE/ACM International Conference on, IEEE, 2010, pp. 41–48.

[33] C. Jeong, T. Ha, J. Hwang, H. Lim, J. Kim, Mars: measurement-based allocation of vm resources for cloud data centers, in: Proc. of Student workhop, ACM, 2013, pp. 63–66.

[34] A. Beloglazov, R. Buyya, Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers, Concurr. Comput. 24 (13) (2012) 1397–1420.

[35] TagMan, Just one second delay in page-load can cause 7 conversions, 2013, (http://www.tagman.com/mdp-blog/2012/03/just-one-second-delay-in-page-load-can-cause-7-loss-in-customer-conversions/).

[36] S. Jacob, Speed is a killer why decreasing page load time can drastically increase conversions, 2011, (https://blog.kissmetrics.com/speed-is-a-killer/).

[37] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, et al., A view of cloud computing, Commun. ACM 53 (4) (2010) 50–58.

[38] D. Moore, C. Shannon, D.J. Brown, G.M. Voelker, S. Savage, Inferring internet denial-of-service activity, ACM Trans. Comput. Syst. 24 (2) (2006) 115–139.

[39] M. Luo, T. Peng, C. Leckie, Cpu-based dos attacks against sip servers, in: Network Operations and Management Symposium, 2008. NOMS 2008. IEEE, IEEE, 2008, pp. 41–48.

[40] H. Liu, A new form of DOS attack in a cloud and its avoidance mechanism, in: Proc. of 2010 workshop on Cloud computing security, ACM, 2010, pp. 65–76.

[41] Z. Xu, H. Wang, Z. Xu, X. Wang, Power attack: An increasing threat to data centers, in: Proc. of NDSS, vol. 14, 2014.

[42] Z. Anwar, A.W. Malik, Can a DDos attack meltdown my data center? A simulation study and defense strategies, Commun. Lett. IEEE 18 (7) (2014) 1175–1178, doi:10.1109/LCOMM.2014.2328587.

[43] G. Somani, S. Chaudhary, Application performance isolation in virtualization, in: Cloud Computing, International Conference on, IEEE, 2009, pp. 41–48.

[44] D. Gupta, L. Cherkasova, R. Gardner, A. Vahdat, Enforcing performance isolation across virtual machines in Xen, in: Middleware 2006, Springer, 2006, pp. 342–362.

[45] Y. Li, X. Tang, W. Cai, On dynamic bin packing for resource allocation in the cloud, in: Proc. of the 26th ACM Symp. Parallelism in algorithms and architectures, ACM, 2014, pp. 2–11.

[46] T. Wood, P. Shenoy, A. Venkataramani, M. Yousif, Black-box and gray-box strategies for virtual machine migration, in: Proceedings of the 4th USENIX NSDI, Berkeley, CA, USA, in: NSDI, 2007, p. 17.

[47] G. Somani, M.S. Gaur, D. Sanghi, M. Conti, R. Buyya, Ddos attacks in cloud computing: Issues, taxonomy, and future directions, arXiv preprint arXiv:1512.08187(2015).

[48] Z. Xu, H. Wang, Z. Wu, A measurement study on co-residence threat inside the cloud, in: 24th USENIX Security 15, USENIX Association, Washington, D.C., 2015, pp. 929–944.

[49] J. Latanicki, P. Massonet, S. Naqvi, B. Rochwerger, M. Villari, Scalable cloud defenses for detection, analysis and mitigation of DDos attacks, in: Future Internet Assembly, 2010, pp. 127–137.

[50] H. Wang, Q. Jia, D. Fleck, W. Powell, F. Li, A. Stavrou, A moving target DDos defense mechanism, Comput. Commun. 46 (2014) 10–21.

[51] S. Zhao, K. Chen, W. Zheng, Defend against denial of service attack with VMM, in: Grid and Cooperative Computing, 2009. GCC'09. Eighth International Conference on, IEEE, 2009, pp. 91–96.

[52] G. Yossi, H. Amir, S. Michael, G. Michael, Cdn-on-demand: An affordable ddos defense via untrusted clouds, in: NDSS 2016, 2015.

[53] G. Somani, A. Johri, M. Taneja, U. Pyne, M.S. Gaur, D. Sanghi, Darac: Ddos mitigation using ddos aware resource allocation in cloud, in: 11th International Conference, ICISS, Kolkata, India, December 16-20, 2015, Proceedings, 2015.

[54] F. Palmieri, S. Ricciardi, U. Fiore, Evaluating network-based dos attacks under the energy consumption perspective: new security issues in the coming green ICT area, in: BWCCA, International Conference on, 2011, pp. 374–379, doi:10.1109/BWCCA.2011.66.

[55] V. Bhardwaj, A. Sharma, G. Somani, Client-side verifiable accounting in infrastructure cloud, in: Advances in Computing, Communications and Informatics (ICACCI), 2015 International Conference on, IEEE, 2015, pp. 361–366.

[56] V. Sekar, P. Maniatis, Verifiable resource accounting for cloud computing services, in: Proceedings of the 3rd ACM workshop on Cloud computing security workshop, ACM, 2011, pp. 21–26.

[57] K.-W. Park, J. Han, J. Chung, K.H. Park, Themis: A mutually verifiable billing system for the cloud computing environment, Serv. Comput. IEEE Trans. 6 (3) (2013) 300–313.

[58] T. Peng, C. Leckie, K. Ramamohanarao, Survey of network-based defense mechanisms countering the dos and DDos problems, ACM Comput. Surv. 39 (1) (2007), doi:10.1145/1216370.1216373.

[59] C. Douligeris, A. Mitrokotsa, {DDoS} attacks and defense mechanisms: classification and state-of-the-art, Comput. Netw. 44 (5) (2004) 643–666. http://dx.doi.org/10.1016/j.comnet.2003.10.003.

[60] R.C. Chiang, S. Rajasekaran, N. Zhang, H.H. Huang, Swiper: Exploiting virtual machine vulnerability in third-party clouds with competition for i/o resources, IEEE Trans. Parallel Distrib. Syst. 26 (6) (2015) 1732–1742, doi:10.1109/TPDS.2014.2325564.

[61] R. Shea, J. Liu, Understanding the impact of denial of service attacks on virtual machines, in: Proc. 20th International Workshop on Quality of Service, IEEE Press, 2012, p. 27.

[62] R. Shea, J. Liu, Performance of virtual machines under networked denial of service attacks: Experiments and analysis, Syst. J. IEEE 7 (2) (2013) 335–345.

[63] R. Riggio, F. De Pellegrini, D. Siracusa, The price of virtualization: Performance isolation in multi-tenants networks, in: Network Operations and Management Symposium (NOMS), 2014 IEEE, IEEE, 2014, pp. 1–7.

[64] H. Aljahdali, P. Townend, J. Xu, Enhancing multi-tenancy security in the cloud iaas model over public deployment, in: Service Oriented System Engineering (SOSE), 2013 IEEE 7th International Symposium on, 2013, pp. 385–390, doi:10.1109/SOSE.2013.50.

[65] S. Yu, R. Doss, W. Zhou, S. Guo, A general cloud firewall framework with dynamic resource allocation, in: ICC, IEEE, 2013, pp. 1941–1945.

[66] F. Al-Haidari, M. Sqalli, K. Salah, Evaluation of the impact of EDos attacks against cloud computing services, Arabian J. Sci. Eng. 40 (3) (2014) 773–785.

[67] R. Miao, M. Yu, N. Jain, Nimbus: cloud-scale attack detection and mitigation, in: Proceedings of the 2014 ACM conference on SIGCOMM, ACM, 2014, pp. 121–122.

[68] ReviewMyLife.co.uk, Amazon cloudfront and s3 maximum cost, 2011, (http://www.reviewmylife.co.uk/blog/2011/05/19/amazon-cloudfront-and-s3-maximum-cost/).

[69] F. Palmieri, M. Ficco, A. Castiglione, Adaptive stealth energy-related dos attacks against cloud data centers, in: Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2014 Eighth International Conference on, 2014, pp. 265–272, doi:10.1109/IMIS.2014.33.

[70] M. Ficco, F. Palmieri, Introducing fraudulent energy consumption in cloud infrastructures: a new generation of denial-of-service attacks, Syst. J. IEEE PP (99) (2015) 1–11, doi:10.1109/JSYST.2015.2414822.

[71] J. Idziorek, M. Tannian, Exploiting cloud utility models for profit and ruin, in: Proc. IEEE International Conference on Cloud Computing (4th IEEE CLOUD'11), IEEE Computer Society, Washington, DC, USA, 2011, pp. 33–40.

[72] S. VivinSandar, S. Shenai, Economic denial of sustainability (EDos) in cloud services using HTTP and XML based DDoS attacks, Int. J. Comput. Appl. 41 (20) (2012) 11–16.

[73] N. Vlajic, A. Slopek, Web bugs in the cloud: feasibility study of a new form of EDos attack, in: Globecom Workshops (GC Wkshps), 2014, IEEE, 2014, pp. 64–69.

[74] K. Salah, J.M.A. Calero, S. Zeadally, S. Al-Mulla, M. Alzaabi, Using cloud computing to implement a security overlay network, IEEE Secur. Priv. 11 (1) (2013) 44–53.

Gaurav Somani is an Assistant Professor at Department of Computer Science and Engineering, Central University of Rajasthan, India. He has completed his Bachelor of Engineering (BE) in Information Technology from University of Rajasthan with honors and Master of Technology (MTech) in Information and Communication Technology from DAIICT, Gandhinagar India with Distinction. He is pursuing his PhD from Malaviya National Institute of Technology, Jaipur, India. His research interests include Distributes Systems and Security Engineering. He has authored a book/ monograph on Scheduling and Isolation in Virtualization. He has published number of papers in various conferences and journals of international repute like ACM SINCONF, ACM CGC, IEEE CLOUD and Elsevier FGCS and ComNet. He has served as TPC member in multiple International conferences and reviewer of top journals like IEEE transactions on cloud computing. He is a member of IEEE and ACM.

Manoj Singh Gaur is a Professor in the Department of Computer Science and Engineering at Malaviya National Institute of Technology Jaipur, India. He has obtained his Ph.D. from University and Southampton, UK and masters from Indian Institute of Science Bangalore, India. He has supervised research in the areas of Information Security and Networks on Chip. He has published over 150 papers in peer-reviewed reputed conferences and journals. He has coordinated multiple national and international projects in the domains of Information Security and Networks on Chip. He is on the editorial board of CSI Transactions on ICT, CSI journal of computing, and earlier with IET D&T. He has been associate editor of IET Computer and Digital Techniques (SI) and Elsevier JISA. He has been organizing chair of ACM SIN 2012, VDAT 2013, and SPACE 2015. He is program chair of VDAT 2015, IC3, and steering committee of VLSI Design. He is general chair of ICISS 2016. He is a member of IEEE and ACM.

Dheeraj Sanghi is a Professor of Computer Science and Engineering at IIT Kanpur. Since August 15, he has started working with IIIT Delhi. From 2008 - 2010, he served as the Director, LNM Institute of Information Technology (LNMIIT), a public private partnership University in Jaipur. His research interests include network performance optimization, security and distributed systems. He has visited about 70 colleges in India to discuss issues related to careerplanning, future of IT industry, curriculum and various technical/research talks. He has published a large number of papers at reputed International conferences and journals. He regularly writes his popular ideas about higher education and learning on his blog, dsanghi.blogspot.com. Professor Sanghi has a B. Tech from IIT Kanpur, and M. S. and Ph. D. from University of Maryland.

Mauro Conti received his MSc and his PhD in Computer Science (advisor Prof. Luigi V. Mancini) from Sapienza University of Rome, Italy , in 2005 and 2009, respectively. In 2008, he was Visiting Researcher (supervised by Prof. Sushil Jajodia) at the Center for Secure Information Systems (CSIS) at George Mason University, Fairfax, VA, USA. In 2009 he was selected for the ERCIM (European Research Consortium for Informatics and Mathematics) "Alain Bensoussan" Fellowship (currently a EU Marie Curie COFUND action). From 2009 to 2011 he was Postdoctoral Researcher (supervised by Prof. Andrew S. Tanenbaum and Prof. Bruno Crispo) at Vrije Universiteit Amsterdam, The Netherlands. In November 2010, he was visiting researcher at UCLA University of California, Los Angeles, CA, USA (working with Prof. Mario Gerla). In 2011, he joined University of Padua, Italy , (among the best Italian universities) as Assistant Professor (tenured faculty). In the summer of 2012, 2013, and 2014 he was visiting Assistant Professor at UCI University of California, Irvine, CA, USA (working with Prof. Gene Tsudik). From 2012, he is a EU Marie Curie Fellow. In October-November 2013 he was a DAAD Fellow at the Center for Advance Security Research Darmstadt (CASED), TU Darmstadt, Germany (working with Prof. AhmadReza Sadeghi). In 2014, he was elevated to the IEEE Senior Member grade, and in 2015 he became Associate Professor. His research interests are mainly in the area of security and privacy. In this area, he published 100+ papers in topmost international peer reviewed journals and conferences, including IEEE TDSC, IEEE TPDS, IEEE TIFS, ACM TWEB, IEEE TSC, IEEE COMST, ACM CCS, ACM AsiaCCS, ACM WiSec, ACM SACMAT, ACM MobiHoc, ACNS, IEEE ICDCS, and ESORICS. He is Associate Editor for several journals, including IEEE Communications Surveys & Tutorials, and he served as Program Committee member of several conferences, including ACM WiSec, ACM CODASPY, ACM SACMAT, IEEE INFOCOM, IEEE CNS, IEEE PASSAT, IEEE MASS, and ACNS. He was panelist at ACM CODASPY 2011. He was General Chair for SecureComm 2012 and ACM SACMAT 2013, and Program Chair for TRUST 2015, and for the Security Track of IEEE CCNC '16.