DE GRUYTER
OPEN

BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **17**, No 4
Sofia • 2017 Print ISSN: 1311-9702; Online ISSN: 1314-4081
DOI: 10.1515/cait-2017-0040

Mitigation of Distributed Denial of Service Attacks in the Cloud

Wael Alosaimi[1], *Michal Zak*[2], *Khalid Al-Begain*[2], *Roobaea Alroobaea*[1], *Mehedi Masud*[1]

[1]*College of Computers and Information Technology, Taif University, Taif, Saudi Arabia*
[2]*University of South Wales, UK*
E-mails: w.osaimi@tu.edu.sa Michal.zak@southwales.ac.uk k.begain@southwales.ac.uk r.robai@tu.edu.sa mmasud@tu.edu.sa

Abstract: *Cybersecurity attacks resulting in loss of availability of cloud services can have significantly higher impact than those in the traditional stand-alone enterprise setups. Therefore, availability attacks, such as Denial of Service attacks (DoS); Distributed DoS attacks (DDoS) and Economical Denial of Sustainability (EDoS) attacks receive increasingly more attention. This paper surveys existing DDoS attacks analyzing the principles, ways of launching and their variants. Then, current mitigation systems are critically discussed. Based on the identification of the weak points, the paper proposes a new mitigation system named as DDoS-Mitigation System (DDoS-MS) that attempts to overcome the identified gap. The proposed framework is evaluated, and an enhanced version of the proposed system called Enhanced DDoS-MS is presented. In the end, the paper presents some future directions of the proposed framework.*

Keywords: *Information processes, cloud computing, security, denial of service, distributed denial of service attacks, economical denial of sustainability.*

1. Introduction

Security as a word in terms of cloud computing is used with increasing frequency. A survey conducted by (Intel, [17]) proved that almost 9 out of 10 respondents are concerned about security in the cloud. It is clear that confidence in the appropriate security measures to protect cloud user's information and services offered to them by the cloud can have a huge impact of the cloud computing industry.

There are a number of security concerns within this specific area, such as specific legal challenges, virtualization issues or possibility of a breach of privacy. All of these challenges are important for a successful turnover of the respondents who will be not concerned about the security within this environment.

Side by side with security, availability is also very important in the cloud. Cloud users do not have the information within their local machine; however, it is essential to be able to simulate the same behaviour. Availability is a necessary aspect of the security which, unfortunately, became a target to the attackers.

Although there are more security aspects which may pose threats, and this work mentions some of them, the main focus will be narrowed down to the availability challenges of the cloud computing. In the beginning, the attacks that harm the availability are presented including the Denial of Service, the Distributed version of the DoS, known as DDoS and Economical Denial of Sustainability. These attacks are explained in detail in the paper. The paper presents principles of particular attacks, the way of launching these attacks and presents the main variants of mentioned attacks. The current mitigation solutions designed as a defence for protecting availability are presented in the third section. All presented solutions are evaluated with their strong points and weaknesses identified.

After the comparison of current mitigation systems, this work presents a DDoS-Mitigation System (DDoS-MS), which suggested improvements to handle the weak points defined in previous systems. Its architecture, implementation, principle and evaluation are presented.

Based on the evaluation of the proposed framework, further improvements were introduced as a new version called Enhanced DDoS-MS. The work demonstrates the principle and architecture and mechanism of the Enhanced DDoS-MS.

At the end of the paper, future challenges are presented from three perspectives. The first demonstrate the security challenges with the cloud in general. The second identifies future issues in terms of Distributed Denial of Service attacks, and the last perspective describes future challenges within the Enhanced Distributed Denial of Service – Mitigation System.

2. Cloud computing security

Cloud computing is a computing paradigm which involves delivering services and applications to customers on-demand basis through the Internet (Mallikarjuna and Venkata [31]). These applications and services employ huge data centres, owned by Cloud Service Providers (CSPs) around the world. They include high-grade servers connected to create what is known as a “cloud” by hosting web servers and web applications (Shynu and Singh [50]). Cloud has several features that provide it with the ability to serve its customers efficiently such as scalability, flexibility, on-demand and elasticity (Wang et al. [59]).

As a result of having these features, the cloud customers can obtain some benefits directly when they adopt the cloud. The most important benefits are decreasing the cost, boosting storage capacity and decreasing IT overheads and concerns (Ying and Dong [62]).

Cloud services are offered on diverse levels; Infrastructure as a Service (IaaS), Software as a Service (SaaS), and Platform as a Service (PaaS). In addition, cloud services are classified into three chief deployment models according to the type of users that can access them; these models are private cloud, public cloud, and hybrid cloud (Steve [55]).

Cloud computing offers distinct services to its users; however, various aspects that may pose security threats to the cloud user or even the cloud provider, will be discussed in this section. To facilitate presenting these threats, they are classified into the following four groups (ENISA [11]):

- Policy and Organisational Risks include loss of control (Roberts and Al-Hamdani [44]), compliance risk (Sangroya et al. [48]), the portability issue (ENISA [11]), and end of service (Kuyoro, Ibikunle and Awodele [26]).
- Legal Issues including contracts, Service Level Agreements (SLAs) designing and applying (Raja and Ramaiah [40]; Trappler [56]), data location (Sangroya et al., [48]; Rittinghouse and Ransome [43]), data breach (Sangroya et al. [48]; Foster et al. [13]; Kuyoro, Ibikunle and Awodele [26]), and Data Deletion (ENISA [11]; Slack [52]).
- Physical Security Issues (Sitaram and Manjunath [51]; Fortinet [12]; Aslan [4]; Sangroya et al. [48]).
- Technical Risks including Virtualisation Vulnerabilities (Sabahi [46]; Vaidya [57]; Virtualizationadmin [58]; Rouse [45]; Schwartz [49]; Jin, Keller and Rexford [19]), service outages (Zhou et al. [63]; Ramgovind, Eloff and Smith [42]), encryption issues (Sangroya et al. [48]; Intel [17]; Ablett et al. [1]), data level security (Mukundrao and Vikram [35]), job starvation issues (Raju, Swarna and Rao [41]; Vaidya [57]), data segregation (Kuyoro, Ibikunle and Awodele [26]), web application security issues (Heng [14]; Meier et al. [32]), multi-tenancy security (Kaur and Vashisht [21]; Kuyoro, Ibikunle and Awodele [26]; ENISA, [11]), Network Attacks such as Distributed Denial of Service (DDoS), Man in the Middle Attack (MITM), IP spoofing, port scanning (Raju, Swarna and Rao [41]; Khan, Fisal and Hussain [23]).

DoS can be launched on different layers, such as transport layer or application layers. Availability is one of the most important features of any network or service. The Flooding or Denial of Service (DoS) attack affects this feature by preventing legitimate users from accessing the network resources. Adversaries generate DoS attacks by sending a huge amount of requests in order to consume the servers processing power and flood the network capacity (bandwidth). As a result, legitimate users cannot access the network or services despite proper authenticity and the right to access the required services at given time (Liu [30]).

The Cloud needs to be protected from three types of threats that affect the web page in a flooding manner. The first type affects its computational capacity by consuming the system resources. The second type wastes the bandwidth by downloading large files from the web server repeatedly affecting its communication capability while the third type harms its security by using password guessing attacks and SQL Injections (Lin et al. [28]). The flooding attacks against a static web page are generated either from botnet, computer virus or any other open Denial of Service tool (Yatagai, Isohara and Sasase [61]). The flooding can be malicious as a Denial of Service attack or normal phenomenon like flash crowd. In (Xie and Yu [60]), flash crowd on the web is defined as the situation of accessing a popular website by a very high number of users simultaneously causing a surge in traffic resulting in the website becoming unreachable.

It is very important to distinguish the Denial of Service from the normal flash crowd. A Denial of Service event is a result of huge amount of requests suddenly generated by a small group of known and unknown users while a flash crowd event is a result of a huge amount of requests generated by a huge number of legitimate users gradually after a specific social event (Xie and Yu [60]). The focus of this section is on the Denial of Service (DoS) event.

2.1. Distributed denial of service

Distributed Denial of Service (DDoS) is the DoS attack that is launched by several distributed sources simultaneously (Raju, Swarna and Rao [41], Somani et al. [53]). In order to encounter such threat, many countermeasures have been proposed. However, the existing solutions are either neglecting initial verification of the source of packets or providing a mechanism that increase the response time for the legitimate users.

The machines that are used in the attack are usually infected by worms, so their owners do not know that they are participating in a malicious attack. The attacker intends to create a network of devices under his control to ensure the success of his attack against the victim. The attacker starts with penetrating some machines and creating backdoors on them so he can control them for a while. Those machines are called bots (zombies). Now, the attacker has a full control on the bots in order to generate several types of attacks including DDoS. Nowadays, there are some tools that can be used to generate DDoS such as TFN (Choi et al. [8], Osanaiye, Choo and Dlodlo [37]). According to (Chopade, Pandey and Bhade [9]) and (Piscitello [39]), some of the common DDoS attacks will be mentioned in the following subsections with some details about the first type:

2.1.1. SYN flood attack

Transmission Control Protocol (TCP) provides a reliable, error checked and ordered connection. To achieve this goal, it has to establish a connection before the data can be transmitted. Setting up connection is achieved by a three-way handshake as shown on Fig. 1.

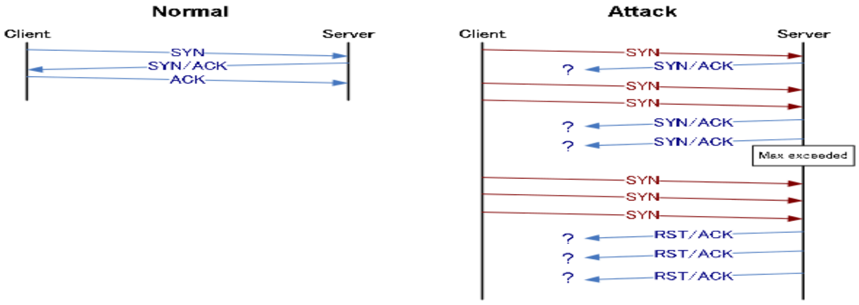


Fig. 1. Establishing the TCP connection (Chen [7])

As soon as a client sends the first initial request namely, the TCP-SYN packet the server has to spend resources to leave the connection open. The server replies with TCP SYN-ACK packet and waits until it receives a final reply from the user namely the TCP ACK packet. Attackers abuse this principle by only sending the TCP-SYN packet without any interest in obtaining reply packets from the server.

Practically the flood happens when an attacker sends a huge number of TCP-SYN packets often with a spoofed IP address. Logically the server has to spend its resources to keep the information about all the clients and leave the connection open in case that the clients do not have a reliable connection. However, the response never comes because the source address is spoofed. (Chopade, Pandey and Bhade [9]) proposed a possible DoS mitigation for the SYN floods by decreasing the server time-out. The logical extension of decreasing server time-outs is to exclude customers with low connection speeds.

2.1.2. Smurf attack

This type of attacks can be created by sending a large amount of moderated ICMP packets with a spoofed IP address to several networks using IP broadcast address. To respond to these requests, all machines in the network will reply to the spoofed address. Therefore, the network works here as a smurf amplifier. If the network is large, it will be overwhelmed by a huge amount of traffic which prevents the legitimate user from accessing the network that is a denial of service. The victim can be affected directly or can pose with other parts of the network a network amplifier of the smurf attack. This overwhelms the network's capacity and prevents the legitimate users from accessing the network services (Peng, Leckie and Ramamohanarao [38]; Chopade, Pandey and Bhade [9]).

2.1.3. Internet Control Message Protocol (ICMP) flood

In this type of attack, the attacker sends successive ICMP echo requests to the targeted machine that must reply with ICMP echo reply to every request. Thus, the huge number of requests and replies causes network disconnection because of the implemented timeouts. Additionally, the ICMP header messages provide the possibility for data injection. This can be exploited for enlarging the whole packet. The types of packets used to generate this attack are called ping packets (Peng, Leckie and Ramamohanarao [38]; Chopade, Pandey and Bhade [9]).

2.1.4. Domain Name System (DNS) amplification attacks

It is clear that the more machines that are involved in the attack, the greater traffic will be generated. Therefore, the impact on the victim would also be bigger. There are ways how to enlarge the DDoS.

The first possibility for how to make the attack stronger is by using the current infrastructure and services which by default give answers after the requests. Legitimate service such as DNS is one of the possibilities for how to increase the efficiency. Attackers can use an extension that allows large messages and from a request that has approximately 60 bytes, the response can enlarge up to 4000 (Piscitello [39]).

2.1.5. Simple network management protocol amplification attacks

Simple Network Management Protocol (SNMP) can also be used as an amplifier. The principle is also the same as in the previous case. The botnet network which includes a number of machines, under the control of an attacker will send a request to a network gateway. However, instead of sending a response to the real botnet machine, it will send a reply to the victim IP address which was injected into the packet as the source IP address. The SNMP request is again smaller than the actual response (Peng, Leckie and Ramamohanarao [38]; Chopade, Pandey and Bhade [9]).

2.2. Economical DDOS

In the cloud computing era, a new type of DDoS attack called Economical Denial of Sustainability (EDoS) was introduced by (Hoff [15]). EDoS is "packet flood that stretches the elasticity of metered-services employed by a server, e.g., a cloud-based server" (Khor and Nakao [24]).

An EDoS attack can be generated by distantly running bots to flood the targeted cloud service using faked requests that are hidden from the security alarms. Therefore, the cloud service will be scaled up to respond to the on-demand requests. As the cloud depends on pay-per-use base, the user's bill will be charged for these faked requests, causing the user to withdraw from the service (Sqalli, Al-Haidari and Salah [54]). In the end, the cloud provider will lose its customers, as they will believe that an on-premise data centre is better and cheaper for them than the cloud, that forces them to pay for services they did not request (Hoff [15]; Kumar et al. [25]).

EDoS attack is a Distributed Denial of Service (DDoS) attack with a different impact. Traditional DDoS attack aims to overwhelm the servers and the bandwidth of a network or a website in order to make them unavailable to their intended users. It is hard of DDoS attack to harm the cloud resources in the same way as the cloud has a huge pool of resources. However, the attackers can generate the DDoS attack against the cloud customer's network. In this scenario, a huge amount of faked requests will be sent to the customer's system which- under cloud contract- will be served by the cloud provider. Hence, the provider can scale up the required infrastructure of the customer in response to its high demand. This process will be reflected in the customer's bill. So, the customer will find that the cloud is not affordable. Spreading the same feeling among many customers will affect the providers' profit. The network security attacks are classified by (Newman [36]) into two types, harmful and costly. Based on this, it is clear that the DDoS attack is a harmful attack while the EDoS is a costly one.

Based on what is mentioned above, the solution that can encounter DDoS attacks against a cloud customer's network must apply a proactive method in such way that the cloud providers can protect their edges that are their customers' networks from EDoS attacks.

There are a number of methods proposed to encounter these attacks. However, these methods are either testing all packets coming from the source causing end-to-end latency or testing the first packet only without any other verification process which is not enough to protect the system. Limiting the end-to-end latency is a very important feature besides providing a robust defence system against the malicious attacks. In (Newman [36]), the authors emphasize the importance of such aspect as the organisations must provide a balance between the security and convenience for their users. Designating a maximum threshold for the customer usage to ensure that the customer's bill does not exceed their satisfied limit in order to prevent the EDoS impacts is not acceptable under the cloud concept. The service can be considered as a cloud service if and only if it is scalable and elastic, is metered by use, has broad network access, has shared pool of resources, and provided as an on-demand self-service (Mell and Grance [33]).

3. Current mitigation techniques

The Distributed Denial of Service (DDoS) countermeasure techniques are divided into two types: reactive and proactive (Beitollahi and Deconinck [6]).

The reactive method such as the filtering system is waiting for an attack to occur and then tries to mitigate its impact. On the other hand, proactive solutions such as the overlay-based techniques involve treating the source of packets before reaching the protected server. These techniques include other components besides the filters. They depend on distributed firewalls or nodes in order to hide the location of the protected server (Morein et al. [34]; Beitollahi and Deconinck [6]; Kumar et al. [25]).

Six existing frameworks will be presented for DDoS mitigation, which are SOS, Kill-Bots, WebSOS, Fosel, CLAD and DaaS.

3.1. Secure Overlay Services (SOS)

The Secure Overlay Services (SOS) is proposed by (Keromytis, Misra and Rubenstein [22]). The authors of (Lakshminarayanan et al. [27]) stated that SOS is the first framework to use overlay techniques to indirect the received packets by the target network besides hiding the location of the target server in order to encounter the denial of service attacks. It aims to allow communication between an authenticated user and the victim server. Authentication of the user means that the server gives prior consent to this user to access the network. It consists of a set of nodes that are classified into four groups. The first group is the Secure Overlay Access Points (SOAP), while the second group is the overlay nodes which connect SOAP nodes with the third group, that is, Beacon nodes. The last group is the Secret Servlets. This reduces the possibility of harmful attacks by applying the filtering process at the edges of the protected network and by providing anonymity and randomness to the architecture, thus making it difficult for an attacker to affect the nodes along the path to the target. SOS uses a large number of overlay nodes that are considered as distributed firewalls to augment the survivability by increasing the amount of resources the attacker must spend to successfully affect the connectivity of legitimate users (Keromytis, Misra and Rubenstein [22]).

SOS employs static routing via the chord overlay network and several servlet nodes in case of fault tolerance. Adversaries are glad about this mechanism of SOS because their brute force method can detect a servlet node in a faster manner. Detecting just one of these servlet nodes is sufficient to overwhelm the target server with a flooding attack. The attackers can achieve this detection of a servlet node by monitoring the traffic of a legitimate user passively (Beitollahi and Deconinck [5]).

3.2. Kill-Bots

Kill-Bots, which is a kernel extension to protect web servers from application-layer DDoS attacks. It authenticates the clients by using graphical tests (CAPTCHA). It distinguishes the zombies from the human users who are unwilling or unable to solve the test by observing the behavior of the user who failed to pass the test. The user is considered as a zombie and therefore forbidden from accessing the server if it continuously sends successive requests to the server despite repeated failures in passing the test. Moreover, Kill-Bots modifies the 3-way handshake process of the TCP connection to protect the authentication technique from DDoS attacks by not creating a new socket upon the end of TCP handshake process (Kandula et al. [20]).

3.3. WebSOS

Morein et al. [34] presented an approach called WebSOS. It has the same architecture as SOS but differs from it in some aspects of its implementation. Legitimate clients can access the web servers during the DoS with this implementation. The architecture employs a mixture of packet filtering, consistent hashing, Graphic Turing Tests (GTT), overlay networks, and cryptographic protocols for data origin authentication to offer services to the casual web browsing user [34].

3.4. Fosel

It is a proactive solution against DoS attacks. It is called Filtering by helping an **overlay security layer (Fosel).** It is composed of firewalls, an overlay network with secret green nodes, and a specific filter called Fosel filter in front of each protected server. Fosel technique aims to protect the target by using the Fosel filters that accept only the approved packets by the green nodes and drop the other packets. As a result, the filter cannot be a victim of an attack that resulting from spoofing the sources IP addresses. Fosel technique is simple as there is no need to notify and then modify the filter if the application site's location is changed according to the filter’s independence from sites location. Moreover, the adversary cannot employ a spoofed IP address in generating attacks against the target (Beitollahi and Deconinck [6]).

3.5. CLAD

CLoud-based Attack Defence system (CLAD) aims to protect web servers from flooding attacks by providing a security system in the form of a network service working on a huge cloud infrastructure which is considered as a supercomputer. Therefore, this supercomputer can defeat network layer attacks against any CLAD node which can be a virtual machine or application that is running web proxies. CLAD consists of a DNS server and a group of CLAD nodes. Every CLAD node can be considered as a web proxy that has many control measures such as congestion control measures, pre-emption, authentication, admission control, and network layer filtering.

The protected server that can be a single server or a set of servers must be hidden from the public and only accepts traffic from the CLAD nodes. The protected server

IP address is known only to the CLAD nodes, so the DNS server replies to any request from the Internet with an IP address of a CLAD node (Du and Nakao [10]).

3.6. DaaS

DDoS mitigation as a Service (DaaS) tackles the DDoS problems by creating a metered pool that has more resources than the botnets to facilitate the harnessing of idle resources from current or future services without alteration. DaaS framework aims to hide the details of the framework, enabling using the framework by the clients and servers without any modification, granting traffic control reception to the server, and enabling any system to be employed as an intermediary. It depends on using SSL certificate with the public key, crypto puzzles, and DNS server. DaaS consists of intermediary plug-ins, multiple stacks, accounting unit and a self Proof of Work (sPoW) consists of a puzzle generator, puzzle requesters, puzzle distributors, and a connection manager (Khor and Nakao [24]).

4. EDoS countermeasures

There are a number of frameworks have been proposed to protect the cloud from the EDoS attacks. Four of these solutions will be presented. They are four.

4.1. EDoS-Shield framework

This framework is proposed by (Sqalli, Al-Haidari and Salah [54]). Its main idea is to check if the requests are generated from botnets or legitimate users. The EDoS-Shield approach is verifies only the first packet by applying CAPTCHA verification, and then accepts or denies the subsequent packets from the same source that has the same IP address. It is a milestone in the techniques that used as solutions, as the authors have focused on solving the end-to-end latency issue.

The main parts of the EDoS-Shield architecture are Virtual Firewalls (VF) and a cloud-based overlay network called Verifier Nodes (V-Nodes). The virtual firewall works as a filter with white and black lists that store the IP addresses of the originating sources. The verifier node verifies the sources using graphic Turing tests such as CAPTCHA to update the lists according to the verification process results. The virtual firewall can be applied as a virtual machine that can filter and route the packets. The white list stores the authenticated source IP addresses so their following traffics will be allowed to pass the firewall filtering mechanism and access the protected system. On the other hand, the black list stores the unauthenticated source IP addresses so their following traffics will be dropped. The two lists must be updated periodically (Sqalli, Al-Haidari and Salah [54]).

4.2. In-Cloud EDDoS Mitigation Web Service (Scrubber Service) eDDoS mitigation service

This framework has been introduced as an on-demand service. It depends on the In-Cloud Scrubber Service that generates and verifies the Client puzzles (crypto puzzles) used at two different levels of difficulty according to the type of attack against the protected system to authenticate the clients. The user must solve the crypto puzzle by brute force method. The system can be switched either to suspected mode or normal mode. The service provider selects the mode depending on the type of attack against its network. In the suspected mode, an on-demand request is required to be sent to the In-Cloud eDDoS mitigation service (Kumar et al. [25]).

4.3. The enhanced EDoS-Shield framework

Al-Haidari, Sqalli and Salah [2] proposed the Enhanced EDoS-Shield framework as an improvement on their EDoS-Shield framework to mitigate EDoS attacks originating from spoofed IP addresses. They made use of the Time-To-Live (TTL) value found in the IP header to facilitate detecting the IP spoofed packets. As a result of using TTL, this framework avoids refusing a request coming from a source registered on the blacklist. Instead of this, it tests the packet as it may be initiated from a victim of a previous IP address spoofing attempt. Therefore, it prevents DoS attacks on legitimate users, even if their IP addresses have been exploited.

A similar architecture to the EDoS-Shield framework is used. However, to enhance the EDoS-Shield framework by enabling it to mitigate the spoofing attacks that affect its original version, the authors added three additional fields that can be monitored and stored in the white and black lists with their correspondent IP addresses. These fields are the TTL values, a counter of unmatched TTL values in both lists, and the time stamp (attack start time) in the black list (Al-Haidari, Sqalli and Salah [2]).

4.4. Sandar and Shenai framework

It is a framework that relies on a firewall, which works as a filter. The framework consists of a firewall and a client puzzle server. The firewall receives the request from the client and redirects it to a puzzle-server. The puzzle-server sends a puzzle to the client, who either sends a correct or wrong answer of the puzzle. If the answer is correct, the server will send a positive acknowledgment to the firewall that will add the client to its white list, and will forward the request to a protected server to get services. Otherwise, the firewall will receive a negative acknowledgment and put the client in its blacklist (Sandar and Shenai [47]).

5. Existing solutions evaluation

After browsing the DDoS and EDoS countermeasures, there is a need to compare their performance based on the determined aspects, which are verifying the packets with more than a method, protecting the scalability by user's rate limiting, and decreasing the end-to-end latency. So, Table 1 compares the DDoS and EDoS-Countermeasures according to the features that designated at the beginning of this section.

Hence, it is noticed that the existing techniques focused on some aspects and ignored or failed to meet the requirements of others. Therefore, a new framework is designed by the author in a way that considers the above features in order to fill this gap.

Table 1. Comparison between the previous frameworks' performances

Framework	Strong authentication	Protecting the scalability	Decreasing the end-to-end latency
SOS	☒	☒	☒
WebSOS	☒	☒	☒
Fosel	☒	☒	✓
CLAD	☒	☒	☒
DaaS	☒	✓	☒
EDoS-Shield framework	☒	☒	✓
Sandar and Shenai framework	☒	✓	☒
Enhanced DDoS-Shield Framework	☒	☒	☒
In-Cloud eDDoS Mitigation Web Service (Scrubber Service) framework	☒	✓	✓

6. The DDoS-MS framework

The investigation of the existing countermeasures shows that presented mitigation techniques are not sufficient. The improved countermeasure should allow strong verification of the source of the traffic, cloud scalability and minimize the end-to-end latency. The proposed framework is designed to fill this gap. The framework is aware of previous work; therefore it includes the strong aspects of previous systems and improves the weak points. The contribution in this work is providing a proactive protection of the cloud provider on their customer's networks from the economic effects of the DDoS attacks by using a new security technique, which fulfil above mentioned criteria. Moreover, for users, which were verified, it will decrease the end-to-end latency for the legitimate users. The proposed framework is called DDoS-Mitigation System (DDoS-MS).

The architecture of the framework consists of six main components. They are a firewall (or virtual firewall), a verifier node(s), a client puzzle server, a DNS server, green nodes in front of the protected server(s), and a filtering router.

The clear purpose of the firewall is to filter the traffic that comes from users. So, malicious users' traffic will be dropped, and the real users' packets will be released through. The firewall has white and black lists for the sources of packets depending on the result of the verification process, which relies on the verifier node and puzzle server.

Green nodes hide the location of the protected server, and the server does not receive any packet except the packet that is forwarded by these nodes through the filtering router. The router forwards the packets that are coming from the green nodes only, and rejects any other packet. Fig. 2 shows the framework's architecture.

DDOS-MS idea is to test two packets coming from any source in two stages. The former is done by the verifier node(s), which use the Graphical Turing Test (GTT) in verifying the packets. The latter is performed by the client puzzle server, which uses a crypto puzzle to verify the source of packets.

The two-phase testing is designed to be completely randomized. The first phase will happen immediately after receiving the first packet by the firewall. However, the second packet that is chosen for the second test will be randomly selected within an early time of communication to not give the possibility to attacker to prepare the attack for it.

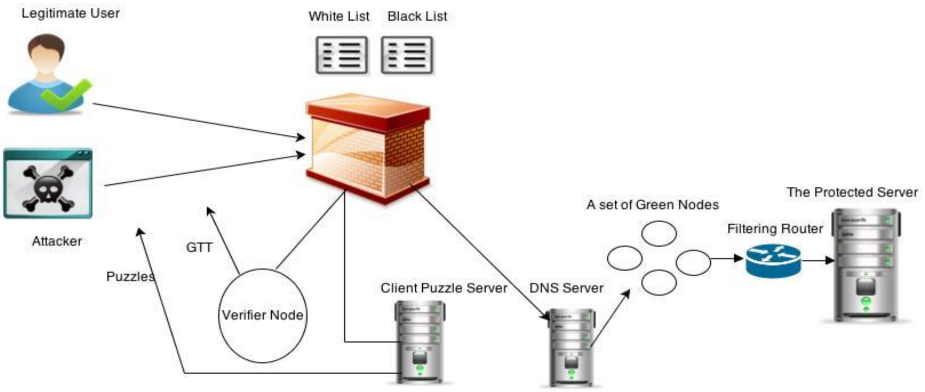


Fig. 2. DDoS-MS architecture

The overview of how are incoming packets handled is described in Fig. 6. Active elements are filled by colour; these elements cooperate with the firewall in the decision-making process of giving access to a particular user. First request symbolizes the first contact of a particular user captured by the firewall; more precisely it is a request that is sent to the Verification Node (VN). Edges represent conditions on which basis will happen the actions in white rectangles. When the verification node verifies the source, the firewall will be informed by positive reply (in Fig. 6 it is a sign +) and opposite by sign (–). Subsequent requests will be managed as it is shown in Fig. 3 under the Following requests.

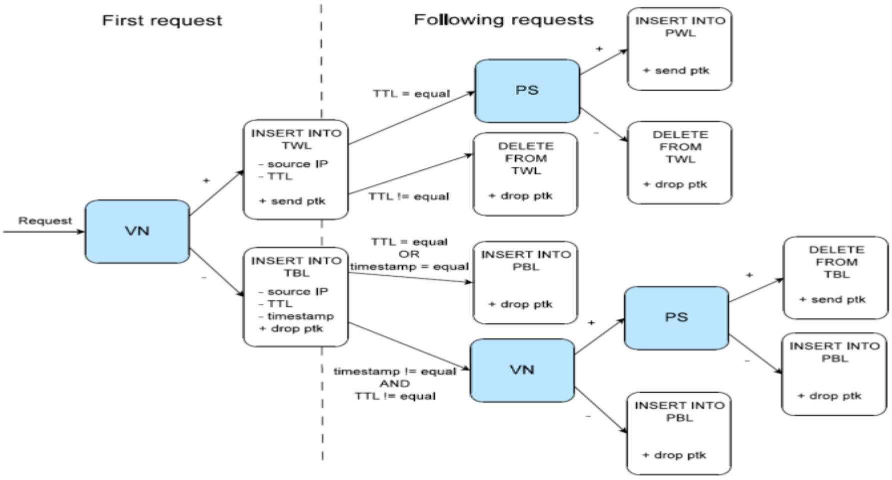


Fig. 3. Framework actions

The graphical representation of framework behaviour is based on framework scenarios (Alosaimi and Al-Begain [3]). The proposed framework is based on the following assumptions in order to limit its scope:

1. The framework must be used in the customer's system and can be used in the provider's system.

2. The attacker's target is to generate DDoS attacks against the cloud to affect its pay-per-use model by exploiting the vulnerabilities in the customer's authentication system.

3. The framework tests two packets which come from any source, assuming those sources' IP addresses are static and the packets are not fragmented, so the TTL (Time To Live) values will not change according to the several paths the fragmented packets can use to reach to the destination.

The idea behind testing only two packets is to enhance the EDoS-Shield framework advantage in the decrement of the end-to-end latency. The role of the verifier node is to verify the sources and distinguishes the legitimate client from the botnets. The second verification, which is performed by the puzzle server, is confirming the legitimacy of the source and strengthening the verification process.

6.1. How DDoS-MS is different from other existing solutions

The previous solutions suffer from verification problems such as SOS and Fosel or from protecting the scalability of the cloud that appears in Enhanced EDoS-Shield, and also from increasing the end-to-end latency as in DaaS framework. This challenge begins when systems are verifying all packets within the flow. Otherwise, when frameworks tests only one packet which is not enough to protect the network from the DDoS and EDoS attacks. The novelty of the DDoS-MS framework lies in focusing on all three objectives; protecting the cloud from DDoS attacks, which implies strong verification process, protects the scalability advantage of the cloud and decreases the end-to-end latency. The framework aims to manage all three challenges at the same time.

The first goal of DDoS-MS is achieved by testing the first two packets. The purpose of the first test is to differentiate between the human user and the botnets that can be employed by attacker to generate DDoS attacks. The human user can be legitimate or malicious. Therefore, the second test is performed to check this. While the client puzzles are usually used in the case which the network is under attack, they are used in this proposed framework as a proactive test as the authors advocated that DDoS-MS is a proactive method to encounter DDoS attacks.

The second objective is achieved by the design of the framework, which gives the customer with a number of users access to the protected servers in the cloud without restriction on scalability.

The third objective of the proposed framework can be achieved by testing only two packets from any source. Our framework does not check all the packets which are received by network interface of the firewall; the end to end latency is decreased by the checking only two random packets from a particular user. Therefore, the legitimate users can get their requested services quickly and without repeated tests.

One more difference between this solution and others, which is using the expression (remove the packet's IP address from the white list or black list) in order to be more resilient. It is used in two cases:

1. When the user passes the GTT test and his/her IP address is recorded in the white list: If he/she fails in the second test using puzzles, the framework appreciates his/her first successful attempt. So, the framework does not transfer his/her IP address from the white list to the black list. Instead of this, it just removes his/her details from the white list.

2. When the user fails in the GTT test and his/her IP address is recorded in the black list: If he/she passes the second GTT test and puzzle test, the framework does not ignore his/her first failed attempt. So, the framework does not transfer his/her IP address from the blacklist to the white list. Instead of this, it just removes his/her details from the black list.

Evaluation of DDoS-MS was based on real testing, which took place in topology that is possible to see on Fig. 4.

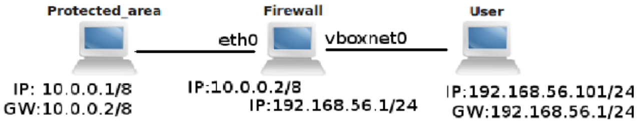
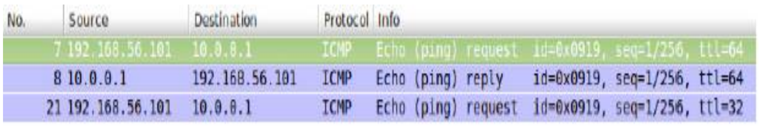


Fig. 4. Topology of testing environment

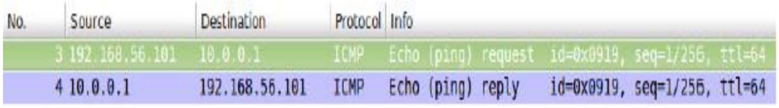
Testing environment consists of a network with three computers, with LINUX platforms. All forwarding was purely based on the firewall; all other possibilities were disabled or forced to not forward anything. Generating packets was done by PackETH (Jemec [18]). Protected area is demonstrated as IP address 10.0.0.1/8, while the entering point which is a firewall is presented as IP address 192.168.56.1/24. For testing purpose, a scenario where user wants to access the protected area with ICMP messages was chosen, more precisely Ping request were sending from the outside world, through the DDoS-MS to the protected area. Fig. 5 shows a Wireshark snapshot of firewall, on the outside interface. One should notice that the same IP address suddenly change TTL from value 64 to 32. In this particular configuration, where the path of the packet in not changed, it suggests an attacker, which spoofed the user’s IP address.



No.	Source	Destination	Protocol	Info
7	192.168.56.101	10.0.0.1	ICMP	Echo (ping) request id=0x0919, seq=1/256, ttl=64
8	10.0.0.1	192.168.56.101	ICMP	Echo (ping) reply id=0x0919, seq=1/256, ttl=64
21	192.168.56.101	10.0.0.1	ICMP	Echo (ping) request id=0x0919, seq=1/256, ttl=32

Fig. 5. Snapshot of firewall on interface, which is connected to the outside world

Taking deeper look, the third packet, which came from the same IP address within the same context, was not able to pass the firewall. By the verification process, it was successfully determined that the packet came from attacker.

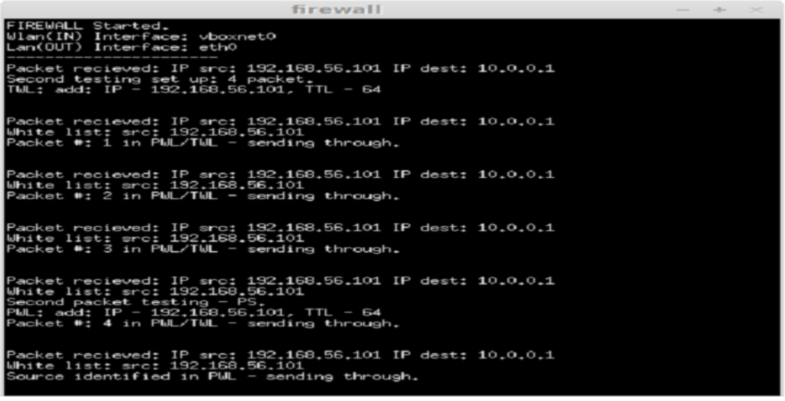


No.	Source	Destination	Protocol	Info
3	192.168.56.101	10.0.0.1	ICMP	Echo (ping) request id=0x0919, seq=1/256, ttl=64
4	10.0.0.1	192.168.56.101	ICMP	Echo (ping) reply id=0x0919, seq=1/256, ttl=64

Fig. 6. Snapshot of firewall on interface, which is connected already inside the protected area

Firewall eliminated the packets from the attacker and let go through just the packets which proved that they were send from the user. Any packets that did not pass this condition were dropped on the outside interface. Therefore, the malicious traffic did not enter the protected area, which is shown in Fig. 6. The firewall of the DDoS-MS is a command line based C^{++} program, which uses Pcap (libpcap0.8 v 1.3.0) network library. The firewall has an option for displaying the current traffic into the terminal, which is possible to see in Fig. 7.

Administrator can see each packet, which is managed by the entry point of DDoS-MS. Each packet will be described with source IP address, destination IP address and TTL value of the packet. Firewall will also inform the administrator about the status of the packet. If the packet came from the user, and it was then classified as non-malicious traffic and, therefore, placed into Temporary While List (TWL) or the Permanent White List (PWL) itself. In the case of attack, the firewall will inform the administrator about it in the same manner.



```
FIREWALL Started.
Wlan(IN) Interface: vboxnet0
Lan(OUT) Interface: eth0
----------------------
Packet recieved: IP src: 192.168.56.101 IP dest: 10.0.0.1
Second testing set up: 4 packet.
TWL: add: IP - 192.168.56.101, TTL - 64

Packet recieved: IP src: 192.168.56.101 IP dest: 10.0.0.1
White list: src: 192.168.56.101
Packet #: 1 in PWL/TWL - sending through.

Packet recieved: IP src: 192.168.56.101 IP dest: 10.0.0.1
White list: src: 192.168.56.101
Packet #: 2 in PWL/TWL - sending through.

Packet recieved: IP src: 192.168.56.101 IP dest: 10.0.0.1
White list: src: 192.168.56.101
Packet #: 3 in PWL/TWL - sending through.

Packet recieved: IP src: 192.168.56.101 IP dest: 10.0.0.1
White list: src: 192.168.56.101
Second packet testing - PS.
PWL: add: IP - 192.168.56.101, TTL - 64
Packet #: 4 in PWL/TWL - sending through.

Packet recieved: IP src: 192.168.56.101 IP dest: 10.0.0.1
White list: src: 192.168.56.101
Source identified in PWL - sending through.
```

Fig. 7. DDoS-MS firewall with the showing traffic option enabled

The early stage of the evaluation shows that this framework can be effective against random attackers, but not persistent attackers such as the hacktivists who can employ a huge number of volunteers to bypass the two tests at the beginning, and then they can use the volunteers' machines as botnets against the network.

Therefore, the authors improved the DDoS-MS framework to encounter these types of attackers by adding and replacing some components and modifying the framework's mechanism. This improved framework is called the Enhanced DDoS-Mitigation System (Enhanced DDoS-MS). This framework will be explained in the next section.

7. The enhanced DDoS-MS framework

The difference between the previous system and the new one is that the DDoS-MS is improved by adding an Intrusion Prevention System (IPS) device that checks the content of the packets in order to detect any malware components using Deep Packet Inspection (DPI) technology and by replacing the overlay system in front of the protected servers with a Reverse Proxy (RP) server.

The Reverse Proxy (RP) server hides the location of the protected servers, manages the load balance between the protected servers, and monitors the traffic rate in order to detect any potential DDoS attacks against the protected servers by designating a pre-determined threshold value for the number of requests coming from any source. The detection process is based on a pre-determined threshold value according to the number of requests in a specific interval (Lin, Liu and Lien [29]).

With this improvement, only the first packet will be tested by the verifier node while the remaining packets will be monitored by an IPS and an RP. The puzzle server will be in this framework applied only for suspicious users that will be determined by exceeding threshold value in the reverse proxy. In addition, the firewall has two more lists – suspicious and malicious lists. The addresses of the sources of packets are placed on the firewall lists depending on the result of the verification and monitoring processes.

If the IPS detects any malware in the packet, its IP address will be placed on the Malicious List (ML). The last layer of the monitoring process is done by the Reverse Proxy (RP). It detects the suspicious users who try to overwhelm the system by sending a huge number of requests without drawing attention to the previous monitoring layers. In this case, the source of such attempts will be placed on the Suspicious List (SL).

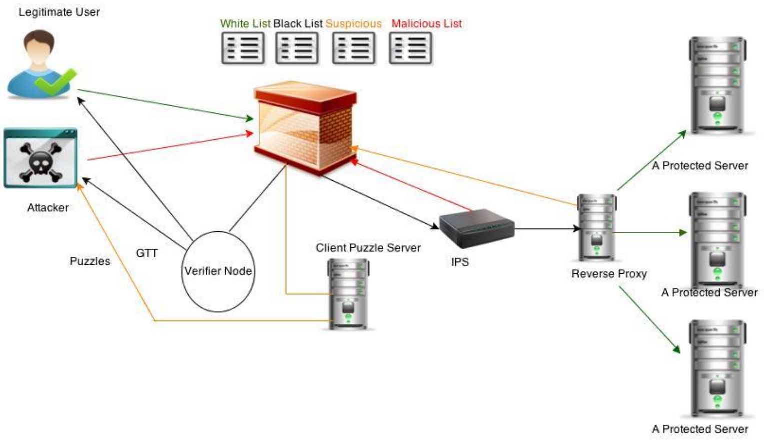


Fig. 8. DDoS-MS architecture

Any packet coming from a suspicious user to the firewall will be forwarded to the client puzzle server which will send a crypto puzzle to its source. The purpose of using the puzzles in this regard is to delay the requests of these suspicious users by consuming a specific time interval and computational power on their side in order to protect the system from the potential DDoS attack. Therefore, the puzzles will be used in this framework as a reactive step, unlike its usage in the DDoS-MS framework. Moreover, it will be used only for suspicious users.

Thus, the legitimate user will not be forced to be tested after passing the verification process neither in the application layer using a GTT test nor in the network layer using puzzles unless his legitimacy is suspected as a result of exceeding the threshold of the traffic rate, or being malicious when their packets contain malware, or changing the packets' TTL values.

The reason for the use of three layers of verification is to distribute the protecting tasks among them and to enable each component to perform a specific security task. Fig. 8 shows the Enhanced DDoS-MS framework's architecture.

8. Conclusion

Despite the distinctive properties which the cloud has, its security aspects must get more attention in order to protect the cloud and maintain its sustainability. There is a need to improve the traditional methods, which are used against attacks as attackers are developing their skills and overcoming most of the existing defence solutions.

The Enhanced DDoS-MS solution which is presented in this paper should be taken as an improvement of previous solutions. It is based on the strong aspects of current mitigation systems and it avoids limitations of these current frameworks in order to produce a robust and effective solution against DDoS and EDoS attacks.

This framework was supported by results from the evaluation of its predecessor framework.

References

1. Ablett, E., D. Bellizzi, J. Byers, S. Cove, M. Dobrusin, A. Frey, J. Hanke. Encryption Advantages and Disadvantages. 2014.
   http://networking116.wikispaces.com/Encryption+Advantages+and+Disadvantages
2. Al-Haidari, F., M. Sqalli, K. Salah. Enhanced EDoS-Shield for Mitigating EDoS Attacks Originating from Spoofed IP Addresses. – In: Proc. of 11th Int. Conf. Trust Secur. Priv. Comput. Commun., IEEE, 2012 [Cited 16 December 2016], pp. 1167-1174.
   http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6296109
3. Alosaimi, W., K. Al-Begain. A New Method to Mitigate the Impacts of Economical Denial of Sustainability Attacks Against the Cloud. – In: Proc. of 14th Annu. Post. Grad. Symp. Converg. Telecommun. Netw. Broadcating. Liverpool John Moores University, Liverpool, UK, 2013, pp. 116-121.
4. Aslan, T. Cloud Physical Security Considerations. IBM Cloud, 2012 [Cited 15 February 2017].
   http://thoughtsoncloud.com/index.php/2012/02/cloud-physical-security-considerations/
5. Beitollahi, H., G. Deconinck. FOSeL: Filtering by Helping an Overlay Security Layer to Mitigate DoS Attacks. – In: Proc. of 7th IEEE Int. Symp. Netw. Comput. Appl., IEEE, July 2008 [Cited 16 December 2016], pp. 19-28.
   http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4579635
6. Beitollahi, H., G. Deconinck. Analyzing Well-Known Countermeasures Against Distributed Denial of Service Attacks. – Comput. Commun., Elsevier B. V., Vol. **35**, Jun 2012, No 11 [Cited 8 October 2016], pp. 1312-1332.
   http://linkinghub.elsevier.com/retrieve/pii/S0140366412001211
7. Chen, E. Y. Detecting TCP-Based DDoS Attacks by Linear Regression Analysis. Signal Process Inf. Technol. – In: Proc. of 5th IEEE Int. Symp., Athens, IEEE, 2005, pp. 381-386.
8. Choi, Y., J. Oh, J. Jang, J. Ryou. Integrated DDoS Attack Defense Infrastructure for Effective Attack Prevention. – In: Proc. of 2nd IEEE Int. Conf. Inf. Technol. Converg. Serv., IEEE, 2010, pp. 1-6.
9. Chopade, S. S., K. U. Pandey, D. S. Bhade. Securing Cloud Servers Against Flooding Based DDOS Attacks. – In: Proc. of Int. Conf. Commun. Syst. Netw. Technol., IEEE, 2013 [Cited 8 October 2016], pp. 524-528.
   http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6524451

10. Du, P., A. Nakao. DDoS Defense as a Network Service. – In: Proc. of IEEE Netw. Oper. Manag. Symp. (NOMS'10), IEEE, 2010, 894-897.
http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5488345

11. ENISA. Cloud Computing Risk Assessment. Eur. Netw. Inf. Secur. Agency, 2009, pp. 9-10.
http://www.enisa.europa.eu/act/rm/files/deliverables/cloud-computing-risk-assessment

12. Fortinet, C. Network and Physical Security in the Cloud. – In: Proc. of Asia Cloud Forum, 2011 [Cited 14 August 2016].
http://www.asiacloudforum.com/content/network-and-physical-security-cloud

13. Foster, I., Y. Zhao, I. Raicu, S. Lu. Cloud Computing and Grid Computing 360. – In: Proc. of Grid Comput. Environ. Work (GCE'08), IEEE, 2008, pp. 1-10.

14. Heng, C. Security Issues in Writing PHP Scripts-And How PHP 4.1.0/4.2.0+ Will Change Your Scripts. 2011.
http://www.thesitewizard.com/archive/phpsecurity.shtml

15. Hoff, C. Cloud Computing Security: From DDoS (Distributed Denial of Service) to EDoS (Economic Denial of Sustainability). – Ration. Surviv., 2008 [Cited 27 September 2016].
http://rationalsecurity.typepad.com/blog/2008/11/cloud-computing-security-from-ddos-distributed-denial-of-service-to-edos-economic-denial-of-sustaina.html

16. Hoff, C. A Couple of Follow-Ups on The EDoS (Economic Denial of Sustainability) Concept. – Ration. Surviv., 2009 [Cited 26 January 2013].
http://www.rationalsurvivability.com/blog/2009/01/a-couple-of-follow-ups-on-the-edos-economic-denial-of-sustainability-concept/

17. Intel. What's Holding Back the Cloud?, 2012 [Cited 26 September 2016].
http://www.intel.com/content/www/us/en/cloud-computing/whats-holding-back-the-cloud-peer-research-report.html

18. Jemec, M. Ethernet Packet Generator. 2014.
http://sourceforge.net/projects/packeth/files/

19. Jin, X., E. Keller, J. Rexford. Virtual Switching without a Hypervisor for a More Secure Cloud. – In: Proc. of 2nd USENIX Work Hot Top Manag. Internet, Cloud, Enterp. Networks Serv. Hot-ICE'12, 2012, pp. 1-6.

20. Kandula, S., D. Katabi, M. Jacob, A. Berger. Botz-4-Sale: Surviving Organized DDoS Attacks that Mimic Flash Crowds. – In: Proc. of 2nd Symp. Networked Syst. Des. Implement, 2005, pp. 287-300.
http://dl.acm.org/citation.cfm?id=1251224

21. Kaur, K., S. Vashisht. Data Separation Issues in Cloud Computing. – Int. J. Adv. Res. Eng. Technol., Vol. **1**, 2013, No X, pp. 26-29.

22. Keromytis, A., V. Misra, D. Rubenstein. SOS: Secure Overlay Services. – In: Proc. of SIGCOMM, ACM, 2002, pp. 61-72.

23. Khan, A., N. Fisal, S. Hussain. Man-in-the-Middle Attack and Possible Solutions on Wimax 802. – In: Proc. of 16j. Int. Conf. Recent Emerg. Adv. Technol. Eng. (iCREATE'09), 2009.

24. Khor, S., A. Nakao. DaaS: DDoS Mitigation-as-a-Service. – In: Proc. of IEEE/IPSJ Int. Symp. Appl. Internet, IEEE, 2011 [Cited 6 November 2012], pp. 160-171.
http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6004147

25. Kumar, M., P. Sujatha, V. Kalva, R. Nagori, A. Katukojwala. Mitigating Economic Denial of Sustainability (EDoS) in Cloud Computing Using In-Cloud Scrubber Service. – In: Proc. of 4th Int. Conf. Comput. Intell. Commun. Networks., IEEE, 2012 [Cited 25 January 2017], pp. 535-539.
http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6375171

26. Kuyoro, S., F. Ibikunle, O. Awodele. Cloud Computing Security Issues and Challenges. – Int. J. Comput. Networks, Vol. **3**, 2011, No 5, pp. 247-252.

27. Lakshminarayanan, K., D. Adkins, A. Perrig, I. Stoica. Taming IP Packet Flooding Attacks. – ACM SIGCOMM Comput. Commun. Rev., Vol. **34**, 2004, No 1, pp. 45-50.

28. Lin, C., C. Lee, J. Liu, C. Chen. A Detection Scheme for Flooding Attack on Application Layer Based on Semantic Concept. – IEEE, 2010, pp. 385-389.

29. Lin, C.-H., J.-C. Liu, C.-C. Lien. Detection Method Based on Reverse Proxy Against Web Flooding Attacks. – In: Proc. of 8th Int. Conf. Intell. Syst. Des. Appl., IEEE, November 2008 [Cited 29 Jun 2016], pp. 281-284.
http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4696475

30. Liu, W. Research on DoS Attack and Detection Programming. – In: Proc. of Third Int. Symp. Intell. Inf. Technol. Appl., IEEE, 2009 [Cited 29 March 2017], pp. 207-210.
http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5368799

31. Mallikarjuna, B., P. Venkata Krishna. OLB: A Nature Inspired Approach for Load Balancing in Cloud Computing. – Cybernetics and Information Technologies, Vol. **15**, 2015, No 4, pp. 138-148.

32. Meier, J., A. Mackman, M. Dunner, S. Vasireddy, R. Escamilla, M. A. Improving Web Application Security Threats and Countermeasures. – MSDN, 2003.
http://msdn.microsoft.com/en-us/library/ff649874.aspx

33. Mell, P., T. Grance. The NIST Definition of Cloud Computing. – National Institute of Standards and Technology, Vol. **53**, 2009, No 6, 50 p.
http://csrc.nist.gov/groups/SNS/cloud-computing/cloud-def-v15.doc

34. Morein, W., A. Stavrou, D. Cook, A. Keromytis, V. Misra, D. Rubenstein. Using Graphic Turing Tests to Counter Automated DDoS Attacks Against Web Servers. – In: Proc. of 10th ACM Conf. Comput. Commun. Secur. (CCS'03). USA, New York, ACM Press, 2003, p. 8.
http://portal.acm.org/citation.cfm?doid=948109.948114

35. Mukundrao, J., G. Vikram. Enhancing Security in Cloud Computing. – Inf. Knowl. Manag., Vol. **1**, 2011, No 1, pp. 40-45.

36. Newman, R. Cybercrime, Identity Theft, and Fraud: Practicing Safe Internet – Network Security Threats and Vulnerabilities. – Int. J. Comput. Appl., ACM, Vol. **9**, 2006, No 12, pp. 11-15.

37. Osanaiye, O., K. Choo, M. Dlodlo. Distributed Denial of Service (DDoS) Resilience in Cloud: Review and Conceptual Cloud DDoS Mitigation Framework. – Journal of Network and Computer Applications, pp. 147-165.

38. Peng, T., C. Leckie, K. Ramamohanarao. Survey of Network-Based Defense Mechanisms Countering the DoS and DDoS Problems. – ACM Computing Surveys, Vol. **39**, 2007, No 1, pp. 1-42.

39. Piscitello, D. Anatomy of a DNS DDoS Amplification Attack. 2011.
http://www.watchguard.com/infocenter/editorial/41649.asp

40. Raja, S., S. Ramaiah. CCDEA: Consumer and Cloud. – DEA Based Trust Assessment Model for the Adoption of Cloud Services, Vol. **16**, 2016, No 3, pp. 52-69.

41. Raju, B., P. Swarna, M. Rao. Privacy and Security Issues of Cloud Computing. – Int. J., Vol. **1**, 2011, No 2, pp. 128-136.

42. Ramgovind, S., M. Eloff, E. Smith. The Management of Security in Cloud Computing. – Inf. Secur., South Africa, IEEE, 2010, pp. 1-7.
http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5588290

43. Rittinghouse, J. W., J. F. Ransome. Cloud Computing: Implementation, Management, and Security. Boca Raton, CRC Press, 2010.

44. Roberts, J., W. Al-Hamdani. Who Can You Trust in the Cloud? – In: Proc. of Inf. Secur. Curric. Dev. Conf. (InfoSecCD'11), ACM Press, 2011, pp. 15-19.
http://dl.acm.org/citation.cfm?id=2047456.2047458

45. Rouse, M. Virtualization Sprawl (VM Sprawl), 2004.
http://whatis.techtarget.com/definition/virtualization-sprawl-virtual-server-sprawl

46. Sabahi, F. Virtualization-Level Security in Cloud Computing. – Communication Software Networks, IEEE, 2011, pp. 250-254.

47. Sandar, V., S. Shenai. Economic Denial of Sustainability (EDoS) in Cloud Services Using HTTP and XML Based DDoS Attacks. – Int. J. Comput. Appl., Vol. **41**, 31 March 2012, No 20, pp. 11-16.
http://research.ijcaonline.org/volume41/number20/pxc3878063.pdf

48. Sangroya, A., S. Kumar, J. Dhok, V. Varma. Towards Analyzing Data Security Risks in Cloud Computing Environments. – ICISTM, 2010, pp. 255-265.

49. Schwartz, M. New Virtualization Vulnerability Allows Escape to Hypervisor Attacks. 2012. **http://www.darkreading.com/risk-management/new-virtualization-vulnerability-allows-escape-to-hypervisor-attacks/d/d-id/1104823?**

50. Shynu, P., J. Singh. A Comprehensive Survey and Analysis on Access Control Schemes in Cloud Environment. – Cybernetics and Information Technologies, Vol. **16**, 2016, No 1, pp. 19-38.

51. Sitaram, D., G. Manjunath. Cloud Security Requirements and Best Practices. – Mov. to Cloud Dev. Apps. New World Cloud Comput., USA, Elsevier, 2012, p. 309.

52. Slack, E. How do You Know That "Delete" Means Delete in Cloud Storage? 2011 [Cited 14 April 2017]. **http://www.storage-switzerland.com/Articles/Entries/2011/8/16_How_do_you_know_that_Delete_means_Delete_in_Cloud_Storage.html**

53. Somani, J., M. Gaur, D. Sanghi, M. Counti, R. Buyya. DDoS Attacks in Cloud Computing: Issues, Taxonomy, and Future Directions. – DDoS Attacks in Cloud Computing. Vol. **107**, pp. 30-48.

54. Sqalli, M., F. Al-Haidari, K. Salah. EDoS-Shield – A Two-Steps Mitigation Technique Against EDoS Attacks in Cloud Computing. – In: Proc. of 4th IEEE Int. Conf. Util. Cloud Comput., IEEE, 2011 [Cited 1 November 2016], pp. 49-56. **http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6123480**

55. Steve, H. Cloud Computing Made Clear. – Bus Week, Vol. **59**, 2008, No 1.

56. Trappler, T. When Your Data's in the Cloud, Is It Still Your Data? Computer World. – Comput. World, 2012. **http://www.computerworld.com/s/article/9223479/When_your_data_s_in_the_cloud_is_it_still_your_data_**

57. Vaidya, V. Virtualization Vulnerabilities and Threats : A Solution White Paper. – Red Cannon Secur. Inc., 2009 [Cited 13 April 2016], pp. 1-7. **http://www.redcannon.com/vDefense/VM_security_wp.pdf**

58. Virtualizationadmin. What is VM Sprawl? 2008. **http://www.virtualizationadmin.com/faq/vm-sprawl.html**

59. Wang, R., X. Sun, X. Yang, H. Hu. Cloud Computing and Extreme Learning Machine for a Distributed Energy Consumption Forecasting in Equipment-Manufacturing Enterprises. – Cybernetics and Information Technologies, Vol. **16**, 2016, No 6, pp. 83-97.

60. Xie, Y., S. Yu. Monitoring the Application-Layer DDoS Attacks for IEEE. – ACM Trans Netw., Vol. **17**, 2009, No 1, pp. 15-25.

61. Yatagai, T., T. Isohara, I. Sasase. Detection of HTTP-GET Flood Attack Based on Analysis of Page Access Behavior. – In: Proc. of IEEE Pacific Rim Conf. Communication Computer Signal Process, IEEE, 2007, pp. 232-235.

62. Ying, L., B. Dong. The Algebraic Operations and Their Implementation Based on a Two-Layer Cloud Data Model, Vol. **16**, 2016, No 6, pp. 5-26.

63. Zhou, M., R. Zhang, W. Xie, W. Qian, A. Zhou. Security and Privacy in Cloud Computing: A Survey. – In: Proc. of 6th Int. Conf. Semant. Knowl. Grids. IEEE, 2010, pp. 105-112. **http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5663489**