

# Short Paper: IP Spoofing Detection for Preventing DDoS Attack in Cloud Computing


Opeyemi Ayokunle Osanaiye

## Cite this paper

Downloaded from [Academia.edu](#) 

[Get the citation in MLA, APA, or Chicago styles](#)

## Related papers

[Download a PDF Pack](#) of the best related papers 



[TCP/IP Header Classification for Detecting Spoofed DDoS Attack in Cloud Environment](#)

Mqhele Dlodlo, Opeyemi Ayokunle Osanaiye

[Techniques and Countermeasures of TCP/IP OS Fingerprinting on Linux Systems](#)

Riaan Stopforth

[DDoS Attacks Detection and Prevention Techniques in Cloud Computing: A Systematic Review](#)

Journal of Computer Science IJCSIS

# Short Paper: IP Spoofing Detection for Preventing DDoS Attack in Cloud Computing

Opeyemi.A. Osanaiye

Department of Electrical Engineering  
University of Cape Town  
Cape Town, South Africa  
e-mail: opyosa001@myuct.ac.za

**Abstract**—Distributed Denial of Service (DDoS) attack has been identified as the biggest security threat to service availability in Cloud Computing. It prevents legitimate Cloud Users from accessing pool of resources provided by Cloud Providers by flooding and consuming network bandwidth to exhaust servers and computing resources. A major attribute of a DDoS attack is spoofing of IP address that hides the identity of the attacker. This paper discusses different methods for detecting spoofed IP packet in Cloud Computing and proposes Host-Based Operating System (OS) fingerprinting that uses both passive and active method to match the Operating System of incoming packet from its database. Additionally, how the proposed technique can be implemented was demonstrated in Cloud Computing environment

**Keywords**—Cloud Computing, DDoS attack, IP Spoofing, OS Fingerprinting.

## I. INTRODUCTION

Cloud Computing was described by [1] as a technology that envelopes some existing related technologies like grid computing, utility computing, cluster computing and distributed computing to provide users with pooled resources as a service.

Cloud service model can be broadly divided into three; Software-as-a-Service (SaaS), Platform-as-a-Service (PaaS) and Infrastructure-as-a-Service (IaaS). It can be deployed to users as Private, Public, Community or Hybrid Cloud [2].

Security issues in Cloud Computing can be viewed from its Confidentiality, Integrity and Availability (CIA). DDoS attack has been identified as the biggest security threat to service availability in Cloud Computing. This prevents legitimate Cloud Users from accessing pool of resources provided by Cloud Providers by flooding and consuming network bandwidth to exhausting computing resources.

IP address spoofing has also been identified as one of the attributes of a DDoS attack, where source IP address is forged.

In this research, different methods for detecting spoofed IP packet during a DDoS attack in Cloud Computing are discussed. Furthermore, a Host-Based OS fingerprinting which uses both passive and active method to match the OS of incoming packet from its database to filter spoofed IP packets is proposed. Additionally, how the proposed technique can be implemented in Cloud Computing environment was demonstrated. The rest of this paper is structured as follows.

Section II describes DDoS attack in Cloud Computing while section III discusses IP spoofing feature in DDoS attack. Section IV introduces OS fingerprinting as a defense mechanism for detecting spoofed IP addresses in DDoS attack. Methodology and implementation are described in Section V and VI while section VII evaluates and analyze the methodology. Finally, the paper is concluded in section VIII.

## II. DDoS ATTACK IN CLOUD

DDoS attack in the Cloud is carried out to overwhelm Cloud resources so as to break them down to the detriment of both the Cloud Providers and the Cloud Users.

This attack can be viewed from its exploitation of weakness of Cloud and flood based. In launching the attack as shown in figure 1, DDoS carries out amplification which can either be a direct or reflection attack.

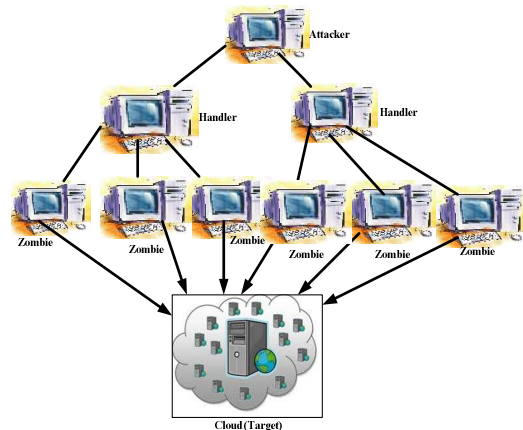


Figure 1. DDoS attack in Cloud

## III. IP SPOOFING IN DDoS ATTACK

DDoS attack is often characterized by spoofing of source IP address to disguise its identity to disallow easy trace back or deceive the Cloud Provider to enjoy certain service accrued to a trusted host. The methods used to detect spoofed IP can be generally classified as either passive or active.

### A. Spoofed IP Packet detection methods

Hop-Count Filtering (HCF) is a method proposed by [3] and was used to filter flooding traffic during a DDoS attack by using the Time-to-Live (TTL) value of the source packet header. The hop count value is compared with the value

obtained from the constructed IP2HC mapping table. Packets will be accepted if the values match while a mismatch will be termed spoofed and rejected. The major drawback of this technique is that all the OS developers do not have the same initial TTL value; therefore mismatch can easily occur during the process of obtaining the initial TTL from the final value.

Path fingerprint method called ANTID (Anti-DDoS) was proposed by [4] for detecting and filtering spoofed packets in DDoS attack. Here, each packet has an embedded unique path fingerprint which identifies the route an IP packet traverses from source to destination. It distinguishes between the different IP paths taken by different IP packets. The server has a map for each of the communicating client and maps the clients IP address to the corresponding path fingerprint. This method is limited in its function as it cannot detect packets with spoofed IP address that does not exist on its mapping.

Trace route is a common network tool that is widely used for determining the path at which a packet traversed. When used to detect a spoofed packet, it will tell the number of hops to the true source of the packet. Among the major disadvantages of this technique is that it is very slow and if the true source is protected by a firewall, the probing packet will return the number of hops to the firewall [5] which will not reflect the hop count of the true source.

TCP interactive method was proposed due to TCP being a connection oriented protocol that ensures reliable delivery of packets by sending ACK messages for every delivered packet between source and destination. On implementing this technique, there will be communication between both sides. This enables detection of spoofed packets as its source will not respond to any probe from the target if it does not exist [5]. Other TCP attributes that can be used includes window size field and the sequence number field of the ACK packet. The major demerit of this technique is that attacker can predict the SYN number value and respond to the target.

This paper proposes the use of Operating System fingerprinting to match the OS of the spoofed IP source to that of the true IP source. This is presented in the next section.

#### IV. OS FINGERPRINTING

OS fingerprinting is the monitoring of an incoming packet to determine the OS the source is running on. It is popularly used by system administrators to identify outdated OS within

their network, locate and patch vulnerable OS and to identify malicious client. OS fingerprinting detection can be either passive or active [6]. It takes the advantage that different OS implements different TCP/IP stack each having its unique signature [6].

##### A. OS fingerprinting features.

In carrying out an OS fingerprint on an IP packet, the advantage that different OS have their unique value combinations for TCP/IP header field is exploited. Among the IP header field attribute that are commonly analyzed are: initial Time to Live (TTL) value, window size, IP DF (Don't Fragment) option and IP ToS (Type of Service) option. These fields will be extracted from the IP header of the incoming TCP SYN or SYN + ACK segment.

##### B. OS fingerprinting methods

OS fingerprinting highly depends on whether it is active or passive. Active fingerprinting involves sending a specially crafted probe packet towards the true source while passive obtains the header features from incoming packets [6].

Common active fingerprinting tools include Nmap, Xprobe2 and SinFP while passive fingerprinting tools includes p0f (Passive OS Fingerprint), OSF (passive OS fingerprinting for iptables) and Ettercap. In this paper we discuss Nmap and p0f.

Nmap is a popular active network mapping tool. It provides scanning feature by sending up to 15 probes which are made up of TCP, UDP and ICMP to open and close ports of the target host. The header fields of the responses are analyzed to identify the OS its running on. This is achieved by matching the observed response to its stored OS database [6].

P0f is a passive fingerprinting technique that was originally written in 2000 by Michal Zalewski [6]. It functions by analyzing the TCP/IP packet header fields to determine the remote host OS. It fingerprints the initiating SYN packet of a remote host connecting to the server and the SYN + ACK response from the server. It defers from Active OS detection tools as it does not send probe packets towards the remote host.

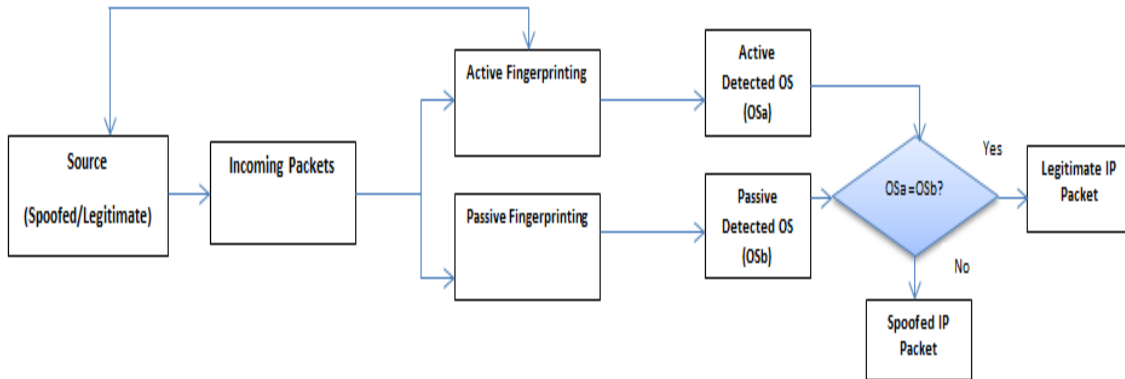


Figure 2. IP Spoofing Detection Block

## V. METHODOLOGY

Active and passive method of OS fingerprint described in figure 2 was deployed. The detector operates in two different modes; the idle mode and the working mode. The detector stays in idle mode unless triggered by high inflow of packets large enough to cause denial of service of cloud resources for legitimate cloud users. During DDoS attack, the incoming malicious packets can be either from a true source or have its IP address spoofed.

During the passive monitoring stage, TCP/IP header features of incoming packets are captured and analyzed using p0f. OS fingerprinting is achieved by matching the analyzed header with p0f's database of known OS to determine the OS. In the active stage, specially crafted probe packet will be sent to the source IP of the received packet using Nmap. If the spoofed IP address is an active address, it will return a response which Nmap captures and uses in identifying its OS from its database.

The observed OS during passive and the probed OS during active will be compared to see if they match. If no ditto, the packet will be classified as spoofed and dropped while a similar match will mean a legitimate IP address.

## VI. IMPLEMENTATION

### Xen Cloud Platform

During implementation, an open source XCP was deployed. XCP 1.6 was installed on a 64bit, 8GB RAM Intel core i5 with 500GB hard disk machine. The XCP hosts 4 VM as described in table I and runs different cloud service for cloud users. Ubuntu 12.04 was the front end that housed Nmap and P0f.

TABLE I: XEN CLOUD PLATFORM 1.6 VM SPECIFICATION

Virtual Machine OS	Product Name	Kernel Version
Centos 6.5	Final	2.6.32
Debian 7.6	Wheezy	3.2
Ubuntu 12.04 LTS	Precise	3.11
Ubuntu 14.04 LTS	Trusty	3.13

## VII. EVALUATION AND ANALYSIS

To evaluate our methodology, ingress and egress connection was established both to the front end of the cloud. Two different scenarios for legitimate user's access and for both legitimate and spoofed malicious users were considered.

*Scenario I:* During connection establishment to the Front end from a remote source, the SYN packet is analyzed by the p0f during the passive stage. It identified the connecting machine as windows XP OS. During the active stage the remote OS was identified as windows XP. In the egress

connection, a similar process took place with p0f identifying the connecting host as a Linux OS 2.6 kernel and Nmap identifying the probed source address as a Linux OS with 2.6.32 kernel version. This is shown in figure 3 below

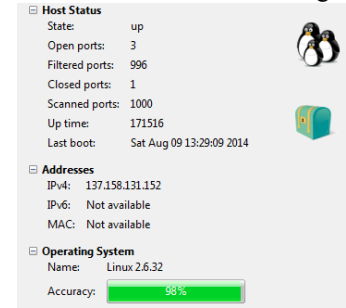


Figure 3. Active Nmap Linux output

*Scenario II:* Both legitimate connection and DDoS attack with Spoofed IP using a traffic generator was launched from outside the cloud environment. During the working mode, the source address of the incoming packets was used to verify the genuine source of the packets. During the verification, the legitimacy of the connection was confirmed by matching OS during passive and active to identify the remote OS as Windows XP as shown in figure 4. Malicious connection with spoofed IP address was also identified with an OS mismatch during the passive and active stage.

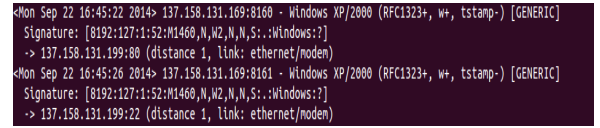


Figure 4. Passive OS fingerprint output using p0f

## VIII. CONCLUSION AND FUTURE WORK

This paper has reviewed different methods used to authenticate the true source of an incoming packet to detect IP spoofing during DDoS attack. We proposed both active and passive host-based OS fingerprinting that verifies the true source of an incoming packet by identifying its OS in Cloud Computing environment.

In future work, the plan is to automate the process for better performance and avoid human intervention.

## REFERENCES

- [1] M. T Khorshed,, A. Ali and S. A. Wasimi, "A survey on gaps, threat remediation challenges and some thoughts for proactive attack detection in cloud computing," *FGCS*, 28(6), pp. 833-851, 2012.
- [2] W.T Tsai, X. Sun and J. Balasooriya, "Service-Oriented Cloud Computing Architecture," *In IEEE Seventh International Conference on Information Technology: New Generations (ITNG)*, Las Vegas, USA, pp. 684-689., 2010.
- [3] J. Cheng, H. Wang and K.G, "Hop-count filtering: an effective defense against spoofed DDoS traffic," *In Proceedings of the 10th ACM conference on Computer and communications security*, USA, pp.30-41. October 2003
- [4] F. Y. Lee and S. Shieh, "Defending against spoofed DDoS attacks with path fingerprint," *Computers & Security*, 24(7), pp.571-586, 2005.
- [5] S. J. Templeton and K. E. Levitt, "Detecting spoofed packets," *In IEEE DARPA Information Survivability Conference and Exposition Proceedings* Vol. 1, pp. 164-175, 2003.
- [6] J. M Allen OS and Application Fingerprinting Techniques, SANS institute InfoSec Reading Room, 2007.