Aalto University
School of Science
Degree Programme in Computer Science and Engineering

Alvaro Garcia

Distributed Denial of Service Attack and OpenFlow:

Improving defense mechanism of DDoS with OpenFlow

Master's Thesis
Espoo, March 15, 2014

DRAFT! — February 6, 2014 — DRAFT!

Supervisors: Assoc. Prof. Aapo Kalliola
Advisor: Assoc. Prof. Aapo Kalliola

Aalto University
School of Science
Degree Programme in Computer Science and Engineering

ABSTRACT OF
MASTER'S THESIS

Author:	Alvaro Garcia		
Title:	Distributed Denial of Service Attack and OpenFlow: Improving defense mechanism of DDoS with OpenFlow		
Date:	March 15, 2014	**Pages:**	14
Major:	Computer Science and Engineering	**Code:**	T
Supervisors:	Assoc. Prof. Aapo Kalliola		
Advisor:	Assoc. Prof. Aapo Kalliola		

The original aim of the Internet was to provide an open and scalable network among research and educational communities, where billions of users are served through a global system of interconnected computer networks.

Since DoS attacks are becoming more commons and emerging new technologies to separate the control plane and the data plane from the network devices (SDN), throughout this survey we will investigate how OpenFlow can help to prevent and locate these kinds of attacks. We will study the different DoS attacks and the current mitigation techniques. We will discuss as well, which of these techniques could be improved with OpenFlow and how develop them. In the end, we will implement and test some identified mitigation techniques and we will study their behaviour.

Keywords:	DoS, DDoS, SDN, OpenFlow
Language:	English

Acknowledgements

Espoo, March 15, 2014

Alvaro Garcia

Abbreviations and Acronyms

2k/4k/8k mode	COFDM operation modes
3GPP	3rd Generation Partnership Project
ESP	Encapsulating Security Payload; An IPsec security protocol
FLUTE	The File Delivery over Unidirectional Transport protocol
e.g.	for example (do not list here this kind of common acronymbs or abbreviations, but only those that are essential for understanding the content of your thesis.
note	Note also, that this list is not compulsory, and should be omitted if you have only few abbreviations

Contents

List of Figures

Chapter 1

Introduction

The original aim of the Internet was to provide an open and scalable network among research and educational communities, where billions of users are served through a global system of interconnected computer networks.

Unfortunately, with the rapid growth of the Internet over the last two decades, the number of attacks on the Internet has also increased rapidly. One of this attacks, consist in disrupt the service provided by a network or server, either crashing the system sending some packets that exploit a software vulnerability or sending a large number of useless traffic to collapse the resources of the service. This kind of attack is known as Denial of Service (DoS) attack, or Distributed Denial of Service attack if is launched for multiple hosts.

There are some design principles of the Internet that facility these kinds of attacks [3]:

Resource sharing: in IP networks, doubt to packet-switched service, users shares all the resources, and one user's service can be disturbed by other user's behaviour, so bandwidth attacks can disrupt service for legitimate users.

Simple Core and Complex Edge: One of the principles of Internet is that the core network should be simple and push all the complexity into the end hosts. That means that the core of the networks is not able to integrate complex application, as authentication, security. Due to this simplification, when an attacker sends packets into the network and the victim receive them, it is almost impossible recognize the real owner of the packets.

Fast Core Networks and Slow Edge Networks: The Core Networks needs to have a high capacity due to the heavy traffic that has to support from many sources to many destinations. In contrast, an edge network needs less capacity because it only needs to support its end users. A disadvantage is that traffic from high-capacity core can crush the slow-capacity edge.

Taking advantage of these principles and their vulnerabilities, have been arising a large number of different DoS and DDoS attacks and, as a result, a parallel growing of defense mechanisms to avoid these attacks. We might consider it like a constant battle between both sides, in which technological improvements are taking an important role on it.

In the current network architecture, the network devices (particular routers) are bundle with a specialized control plane and various features. This vertical integration essentially binds you to whatever software and features are shipped with those particular devices. Software Defined Networking effectively breaks these pieces apart.

SDN is a type of network architecture that separates the network data plane (network devices that forwarding traffic) from the control plane (software logic that controls ultimately how traffic is flowing through the network). OpenFlow [2] is a standard interface defined between the control and forwarding layers of an SDN structure.

One of the reasons to separate the control plane and the data plane is that the software control of the network can evolve independently of the hardware.

A second reason is it allows the network to be controlled from a single high-level software program. The software used to control the network (in our case, POX), even though taking in count that use a high-level programming language, has a lower layer abstraction and increase the difficulty for the Network Programmers. Due this inconvenient, it is time to speak about *Frenetic.Frenetic* is a Network Programming language which gives a high-level abstraction from POX, allowing them direct control over the network. *Pyretic (Python + Frenetic)* is one of the Frenetic family programming languages which provide a domain specific sub-language for specifying dataplane packet processing.

The aim of this survey is how might SDN help us to improve current DDoS defense mechanism. Throughout this project, we will review the main DDoS defense and attack mechanisms and further some algorithms already developed and how could be improve them with OpenFlow. We will test these algorithms on virtual scenarios-through Mininet.

This thesis is structure as follows: The next chapter we will explain the background related with this survey. We talk about the current situation of DDoS attacks and defenses and how OpenFlow works and its structure. In the chapter 3,

Chapter 2

Background

2.1 DDoS attack and defense mechanisms

A denial-of-service attack is characterized by an explicit attempt by attackers to prevent the legitimate users of a service from using that service [1] provided by a network or server. There are two manner to launch this kind of attack. The first approach is overwhelm the network and occupy all the resources of a service sending massive volumes of useless traffic

2.2 OpenFlow

The explosion of mobile devices, server virtualization, security problems and advent of cloud service are among the reasons because the networking industry is beginning to question the traditional network architecture. OpenFlow is intended to solve the problem of assigning resources to users in a easy-way giving them the control plane of the network without disturbing the traffic flows.

In traditional routers and switches, both control plane (high level routing decisions) and data plane (packet forwarding) are embedded in the same device. An OpenFlow Switch separate these two functions (Figure 2.1). The data plane function still resides on the switch, while the control plane is moved to a separate device called Controller (see 2.2.2) that manage the switch and communicate to each other over the Secure Channel (see 2.2.1.2) via the OpenFlow protocol (see 2.2.1.3).

The switch contains *flow tables* 2.2.1.1, which are update through OpenFlow protocol adding, updating and deleting its *flow entries*. When the flow traffic arrives to the switch, it checks if the arrived packets match in the flow table, if so, the action defined in the flow entry is executed. Otherwise, the

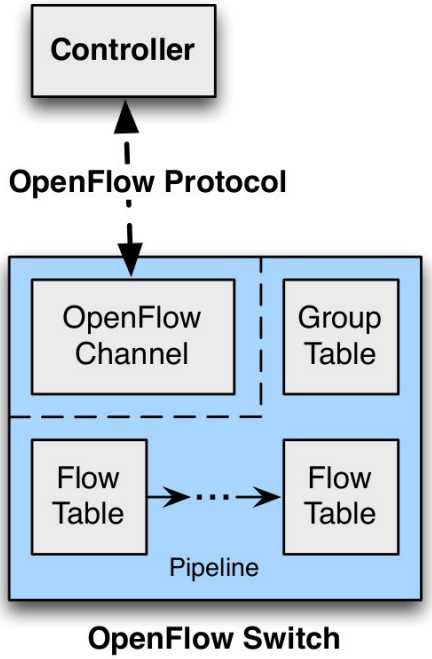

Figure 2.1: OpenFlow Switch components

packet either is send it to the Controller or it is dropped.

Throughout this section, we will explain in detail the main parts of the OpenFlow Switch, as well as the Controller and how they work together. The first version of the OpenFlow (1.1) protocol was released on 2011, one year later, in February 2012, the ONF approved and published the version 1.2. Nowadays, the current version of the protocol and the one that will be used in this project is the 1.4 [4].

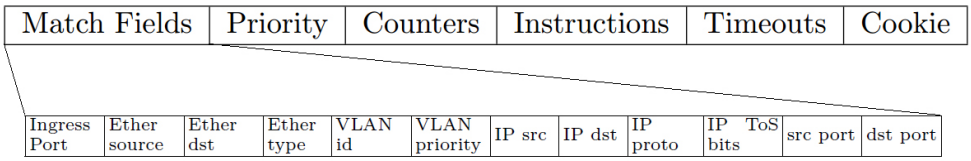

Figure 2.2: Flow entry components and match field headers

2.2.1 Switch Components

2.2.1.1 Flow Table

2.2.1.2 Secure Channel

2.2.1.3 OpenFLow Protocol

2.2.2 Controller

2.3 POX

Bibliography

[1] CC., C. Denial of service attack. `http://www.cert.org/tech_tips/denial_of_service.html`.

[2] OpenFlow. Open Flow standard. http://archive.openflow.org/.

[3] Peng, T., Leckie, C., and Ramamohanarao, K. Survey of network-based defense mechanisms countering the dos and ddos problems. *ACM Computing Surveys (CSUR) 39*, 1 (2007), 3.

[4] Specification, O. S. Version 1.4.0. *Open Networking Foundation* (2013).

Appendix A

First appendix

This is the first appendix. You could put some test images or verbose data in an appendix, if there is too much data to fit in the actual text nicely.

For now, the Aalto logo variants are shown in Figure A.1.

(a) In English

(b) Suomeksi

(c) På svenska

Figure A.1: Aalto logo variants