

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

قال تعالى :

شَهِدَ اللَّهُ أَنَّهُ لَا إِلَهَ إِلَّا هُوَ وَالْمَلَائِكَةُ وَأُولُوا الْعِلْمِ قَائِمًا
بِالْقِسْطِ لَا إِلَهَ إِلَّا هُوَ الْعَزِيزُ الْحَكِيمُ ﴿١٨﴾

صدق الله العظيم

سورة آل عمران الآية رقم ١٨

DEDICATION

This thesis is dedicated to my beloved family for planting the power inside me and uplifting my spirit by supporting me all the way along. Also, dedicated to my friends and my supervisor for spending his time and effort to make this research on its best way.

ACKNOWLEDGMENT

All praise to Allah, today I fold the day's tiredness and the errand summing up between the cover of this humble work. To my brilliant mother and dearest father, to whom they strive to bless comfort and welfare and never stint what they own to push me in the success way.

To my supervisor Dr.Fath Elrahman Ismael Khalifa who supervised, guided, and helped me wholeheartedly.

To the distinguished, collaborator, decent and obliging person Mohammed Widaa who Help me with all his knowledge. Thanks forever

To those who provided to me their knowledge, to my honored teachers, thanks very much.

ABSTRACT

Security is one of the major topics in networking. Nowadays, most companies and organizations spend lots of money on expensive firewalls to enforce security. Software-Defined Networking (SDN) is a new promising technology that can provide cost-efficient solutions with centralized management and programming flexibility. This research work proposes a new approach to network security, which is rapidly developing in modern computer and network industry. By using Software Defined Networking (SDN) with OpenFlow protocol technology, a robust and powerful virtual firewall can be implemented to manage the forwarding behavior of OpenFlow switch. This firewall module can detect and prevent Denial of Service (DoS) attacks and any parallel streams of traffic based on predefined policies and rules that are configurable. This research demonstrates remarkable performance of switch module and firewall module when handling TCP traffic comparing to traditional switch. Firewall module suffers while handling UDP packets due to its security policies and required processes.

المستخلص

الأمن هو أحد الموضوعات الرئيسية في الشبكات. في الوقت الحاضر، معظم الشركات والمنظمات تنفق الكثير من المال على أنظمة جدران الحماية باهظة الثمن لإنفاذ الأمن. الشبكات المعرفة بالبرمجيات هي تقنية واعدة جديدة يمكن أن توفر حلول فعالة من حيث التكلفة مع إدارة مركزية ومرونة في البرمجة. هذا العمل البحثي يقترح نهجا جديدا لأمن الشبكات، الذي يتطور بسرعة في مجال الحواسيب و الشبكات الحديثة. باستخدام الشبكات المعرفة بالبرمجيات مع تقنية بروتوكول أوبن فلو، يمكن تنفيذ جدار حماية ظاهري قوي ومتين لإدارة سلوك توجيه البيانات في الشبكة. وحدة جدار الحماية هذه قادرة على كشف ومنع هجمات إيقاف الخدمة وأي تيارات موازية من حركة البيانات بناء على سياسات و قوانين محددة مسبقا و قابلة للتغيير. هذا البحث يوضح الأداء الملحوظ لكل من نموذج وحدة التبديل ونموذج جدار الحماية عند التعامل مع بيانات بروتوكول التحكم بالإرسال مقارنة مع وحدة التبديل التقليدية. نموذج جدار الحماية يعاني أثناء التعامل مع حزم بروتوكول مخطط بيانات المستخدم بسبب سياسات الأمان والعمليات المطلوبة.

TABLE OF CONTENTS

DEDICATION	II	
ACKNOWLEDGMENT	III	
ABSTRACT	IV	
ABSTRACT IN ARABIC	V	
TABLE OF CONTENTS	VI	
LIST OF TABLES	X	
LIST OF FIGURES	XI	
LIST OF SYMBOLES	XIII	
ABBREVIATIONS	XIV	
CHAPTER ONE	Introduction	2
1.1	Preface	2
1.2	Problem Statement	3
1.3	Proposed Solution	3
1.4	Research Aims and Objectives	4
1.5	Methodology	4
1.6	Thesis Outlines	6

CHAPTER TWO	Literature Review	8
2.1	Background	8
2.1.1	Traditional Firewalls	8
2.1.1.1	Needs of Firewalls	9
2.1.1.2	Firewall Types in Historical Order	9
2.1.2	Software Defined Networking (SDN)	13
2.1.2.1	The Importance of the Separation	16
2.1.3	OpenFlow Protocol	19
2.1.3.1	OpenFlow Switch Components	20
2.1.3.2	OpenFlow Ports	21
2.1.3.3	Flow Table	22
2.1.3.4	OpenFlow Message Types	23
2.1.3.5	Connection Setup	25
2.1.3.6	Multiple Controllers	26
2.1.3.7	Flow Match Fields	27
2.1.3.8	Action Structure	27
2.1.3.9	OF-Config Versions	28
2.1.3.10	OpenFlow Versions	29
2.1.4	SDN Controllers	29
2.1.5	Northbound APIs	31
2.1.6	SDN Use Cases	32
2.2	Related Work	34

CHAPTER THREE	Detection of Network Threats Using SDN	39
3.1	Preparation	39
3.2	Research Activities	41
3.3	Setting up VMware Workstation	42
3.4	Setting up Ubuntu OS	42
3.5	Installing Mininet Emulator	42
3.6	Installing POX controller	45
3.7	Project Design	46
3.7.1	Basic Configuration	46
3.7.2	Firewall Implementation	47
CHAPTER FOUR	Results and Discussion	52
4.1	Mininet without POX Controller	52
4.2	Mininet with POX Controller	53
4.2.1	POX controller Running Hub Module / Switch Module	53
4.2.2	POX controller Running Firewall Module	54
4.2.2.1	First Emulation Scenario	55
4.2.2.2	Second Emulation Scenario	58
4.3	Performance Results	63

CHAPTER FIVE	Conclusion and Recommendations	67
5.1	Conclusion	67
5.2	Recommendations	68
References		70
Appendix A	POXFW1	76
Appendix B	POXFW2	86

LIST OF TABLES

2-1	Required Match Fields	27
2-2	Action Structure	28
2-3	Capability progression of OF-Config	28
2-4	The progression of enhancements to the OpenFlow pipeline from OF v1.1 through OF v1.3	29
2-5	Comparison among the controllers	30

LIST OF FIGURES

2-1	DEC SEAL - First Commercial Firewall	10
2-2	Software-Defined Network Architecture	14
2-3	Control and data plane example implementation	17
2-4	Relationship between components defined in the specification, the OF-CONFIG protocol and the OpenFlow protocol	20
2-5	Main components of an OpenFlow switch	21
2-6	Main components of a flow entry in a flow table	23
2-7	OpenFlow protocol messages	26
2-8	Northbound API	32
3-1	Simplified Project Topology	40
3-2	Flowchart of Research Activities	41
3-3	Installed Packages	45
3-4	Project Topology in Mininet	47
4-1	Mininet Devices and Connectivity Test Result	52

4-2	Mininet Devices Managed by POX controller	53
4-3	Connectivity Test Using Ping tool (With connecting Module)	54
4-4	Connectivity Test Using Ping tool (Without connecting Module)	54
4-5	First POXFW Algorithm	55
4-6	Entropy Value before DoS Attack	56
4-7	Entropy Value after DoS Attack	57
4-8	POXFW Stopping DoS Attack	57
4-9	POXFW Blocked Port 2	57
4-10	Second POXFW Algorithm	59
4-11	POXFW Blocking ICMP Packets From h1 and h2	60
4-12	POXFW Blocking UDP Packets to h1	61
4-13	POXFW Allowing UDP Packets to Other hosts	61
4-14	POXFW Blocking TCP Packets to h4	62
4-15	POXFW Allowing TCP Packets to Other hosts	63
4-16	Comparison in Term of Jitter	64
4-17	Comparison in Term of Throughput	65

LIST OF SYMBOLES

W	Set of Data
n	Number of Elements
X	An Event in a Set
P	Probability of an Event
H	Entropy

LIST OF ABBREVIATIONS

ARP	Address Resolution Protocol
APT	Advanced Packaging Tool
API	Application Programming Interface
ASICS	Application-Specific Integrated Circuits
BSD	Berkeley Software Distribution
CHIPA	Children's Internet Protection Act
COS	Class of Service
CLI	Command Line Interface
DEC	Digital Equipment Corporation
DEC SEAL	DEC Secure External Access Link
DoS	Denial of Service
DDoS	Distributed Denial of Service
FPGA	Field-Programmable Gate Array
FWTK	Firewall Toolkit
FIB	Forwarding Information Base
GUI	Graphical User Interface
IPS	Intrusion Prevention Systems
LAN	Local Area Network
MTU	Maximum Transmission Unit
MAC	Media Access Control

MS-DOS	Microsoft Disk Operating System
MPLS	Multiprotocol Label Switching
NAT	Network Address Translation
NAI	Network Associates Incorporation's
NTP	Network Transport Protocol
ONF	Open Networking Foundation
OVSK	Open vSwitch
OPEX	Operational Expenditure
PPA	personal package archives
POXFW	POX Firewall
RIB	Routing Information Base
SMTP	Simple Mail Transfer Protocol
SDN	Software-Defined Network
SOCKS	Socket Secure protocol
SYN	Synchronize
TTL	Time-to-Live
TCP	Transmission Control Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
TLS	Transport Layer Security
TIS	Trusted Information Systems
UDP	User Datagram Protocol
VM	Virtual Machine
VPN	Virtual Private Networks
WAF	Web Application Firewall