TECHNICAL UNIVERSITY OF MOLDOVA

As a manuscript
UZN: 004.738.5.056.5(043.2)

AMEEN ALI

SECURITY ASSURANCE OF THE COMPUTER NETWORKS BASED ON SOFTWARE DEFINED NETWORK TECHNOLOGIES

Specialty 232.01 Control systems, computers and information networks

PhD Thesis in Engineering Sciences

Scientific Supervizor  **PERJU Veaceslav,**
Dr. hab., Acad. IIA

Author  **AMEEN Ali,**
Eng., M. Sc.

CHIȘINĂU, 2025

UNIVERSITATEA TEHNICĂ A MOLDOVEI

Cu titlu de manuscris
CZU: 004.738.5.056.5(043.2)

AMEEN ALI

ASIGURAREA SECURITATII RETELELOR DE CALCULATOARE BAZATE PE TEHNOLOGII DE REȚEA DEFINITE SOFTWARE

Specialitatea 232.01 Sisteme de conducere, calculatoare si rețele informaţionale

Teza de doctor în Științe Inginerești

Conducător ştiinţific  **PERJU Veaceslav, Dr. hab., Acad. AII**

Autor  **AMEEN Ali, Eng., M. Sc.**

CHIȘINĂU, 2025

ACKNOWLEDGEMENTS

This work is dedicated for all those who helped me in my quest and for all those who helped me in my research works, in the first place I dedicate this work to my god and to my mother, the best and biggest gift I got from God, the reason of everything good in my life and the one I should be writing a whole research about, to my father and brother who stood beside me and of course to my supervisor dr. hab., acad. IIA Veaceslav Perju, who was nice, helpful and well versed in this field who gave me a great deal of help and motivation also, my consultation committee consisting of the esteemed professors: dr. hab., univ. prof. Emilian Guţuleac, dr., assoc. prof. Viorica Sudacevschi, and dr., assoc. prof. Victor Ababii, and all other staff members of Technical University of Moldova who helped make this research done.

TABLE OF CONTENTS

ANNOTATION

To the doctoral dissertation "**Security assurance of the computer networks based on software defined network technologies**" is submitted by Mr. Ali AMEEN for fulfillment of the requirements for the PhD in Engineering Sciences, specialty 232.01– *Control system, computers and information networks.* The dissertation was prepared at the Technical University of Moldova.

The structure of the thesis. The thesis contains Introduction, **3** chapters, general conclusions and recommendations, bibliography of **130** titles. The main text contains **105** pages, includes **47** figures, **23** tables, and **6** annexes. The obtained results of the thesis were published in **13** scientific papers.

Keywords: Software-Defined Networks (SDN), OpenFlow, security, controller, Hydra, virtual private network (VPN), Rivest-Shamir-Adleman (RSA), Blockchain.

Research problem: assuring the security of SDN-based computer networks.

Aim of research is to outline the current security state of the computer networks and to suggest new solutions to patch up different security aspects.

The objectives of thesis include analysis of existing methods and technologies for security assurance of computer networks, elaboration of algorithms and topologies for increasing the security assuring level of computer networks and efficiency evaluation of the proposed topologies for computer network security assuring.

Scientific novelty and originality of the obtained results are reflected in a new framework that assures the security of SDN and uses different techniques combined together to deal with the single point of failure in the SDN architecture, adds a defense mechanism by injecting the attacking source with botnets. And the usage of Petri Nets modeling technique to figure out the best outcome of the proposed topologies and to get different performance parameters that are translated to equations based on that modeling of those specific topologies to determine the one with the best performance in terms of cost saving and deterrence to cyber-threats like DoS/DDoS attacks.

Important scientific solved problem consists in elaboration of a new suite of algorithms and SDN controllers' topologies to increase the security level of SDN and elaboration of the theoretical assessment of computer networks' security level.

Theoretical significance can be described by defining the main problems in security assurance of the computer networks, by specifying the main issues in the SDN paradigm that need to be patched, the theorization of the essential concepts of the proposed algorithms and topologies.

Applicative value of the work is determined by the developed framework, which has a big contribution for the SDN community by proposing new SDN topologies to deal with the centralization issue and by protecting the connection between multiple SDN controllers. Also, provides a better view for the security level of a specific network by measuring it using various mathematical tools that are based on proposed parameters.

Implementation of results. The obtained results were used in Dekart Company's investigations regarding the new approaches in information security.

ADNOTARE

La teza de doctor **„Asigurarea securitatii retelelor de calculatoare bazate pe tehnologii de rețea definite software”** este prezentată de domnul Ali AMEEN pentru conferirea titlului științific de doctor în Științe Inginerești la specialitatea 232.01– Sisteme de conducere, calculatoare şi reţele informaţionale. Teza a fost elaborată la Universitatea Tehnică a Moldovei.

Structura tezei. Teza conține Introducere, **3** capitole, concluzii generale și recomandări, bibliografia din **130** de titluri. Textul de bază constituie **105** de pagini, include **47** figuri, **23** tabele și **6** anexe. Rezultatele obținute ale tezei au fost publicate în **13** lucrări științifice.

Cuvinte cheie: Rețele definite prin software (SDN), OpenFlow, controller SDN, Hydra, rețea privată virtuală (VPN), Rivest-Shamir-Adleman (RSA), Blockchain.

Problemă de cercetare: creșterea nivelului de securitate a rețelelor de calculatoare folosind tehnologii bazate pe SDN.

Scopul cercetării este de a sublinia starea actuală de securitate a rețelelor de calculatoare și de a sugera noi soluții pentru a remedia diferite aspecte de securitate.

Obiectivele tezei includ analiza metodelor și tehnologiilor existente pentru asigurarea securității rețelelor de calculatoare, elaborarea algoritmilor și topologiilor pentru creșterea nivelului asigurării securității rețelelor de calculatoare și evaluarea eficienței topologiilor propuse pentru asigurarea securității rețelelor de calculatoare.

Noutatea științifică și originalitatea rezultatelor obținute sunt reflectate într-un cadru nou care asigură securitatea SDN și folosește diferite tehnici combinate pentru a face față punctului unic de eșec în arhitectura SDN, prevede un mecanism de apărare prin injectarea sursei de atac cu botnetele și utilizarea tehnicii de modelare a rețelelor Petri pentru a determina eficacitatea topologiilor propuse și a obține parametri de performanță, utilizați în ecuații bazate pe modelarea topologiilor specifice în ceea ce privește economisirea costurilor și descurajarea amenințărilor cibernetice precum atacurile DoS/DDoS.

Problema științifică importantă rezolvată constă în elaborarea unui nou set de algoritmi și topologii de controlere SDN pentru creșterea nivelului de securitate al SDN și elaborarea metodologiei de evaluare a nivelului de securitate al rețelelor de calculatoare.

Semnificația teoretică reflectă definirea principalelor probleme în asigurarea securității rețelelor de calculatoare, precizarea principalelor probleme din paradigma SDN care trebuie remediate, teoretizarea conceptelor esențiale ale algoritmilor și topologiilor propuse.

Valoarea aplicativă a lucrării este determinată de cadrul dezvoltat, care reprezintă o contribuție importantă pentru comunitatea SDN prin propunerea a câteva noi topologii SDN pentru a trata problema centralizării și protejarea conexiunii dintre mai multe controlere SDN. De asemenea, oferă o vizualizare mai bună a nivelului de securitate al unei anumite rețele prin măsurarea acesteia folosind diverse instrumente matematice care se bazează pe parametri propuși.

Implementarea rezultatelor. Rezultatele obținute au fost utilizate în investigațiile companiei Dekart privind noile abordări în domeniul securității informațiilor.

АННОТАЦИЯ

К докторской диссертации **«Обеспечение безопасности компьютерных сетей на основе программно-определяемых сетевых технологий»** представленной г-ном Али АМЕЕН на соискание ученой степени доктора наук в области Инженерных наук по специальности 232.01 – *Системы управления, вычислительная техника и информационные сети*. Диссертация была подготовлена в Техническом университете Молдовы.

Структура диссертации: Диссертация содержит введение, **3** главы, общие выводы и рекомендации, библиографию из **130** наименований. Основной текст составляет **105** страниц, включает **47** рисунков, **23** таблиц, и **6** приложений. Полученные результаты диссертации опубликованы в **13** научных работах.

Ключевые слова: программно-определяемые сети (SDN), OpenFlow, контроллер SDN, Hydra, виртуальная частная сеть (VPN), Rivest-Shamir-Adleman (RSA), криптовалюта, блокчейн.

Проблема исследовании: повышение уровня безопасности компьютерных сетей с использованием технологий на базе SDN.

Цель исследовании: определить текущее состояние безопасности компьютерных сетей и предложить новые решения для исправления различных аспектов безопасности.

Задачи диссертации включают в себя анализ существующих методов и технологий обеспечения безопасности компьютерных сетей, разработка алгоритмов и топологий для обеспечения безопасности компьютерных сетей и определение параметров оценки эффективности предлагаемых топологий обеспечения безопасности компьютерных сетей.

Научная новизна и оригинальность полученных результатов отражены в новой структуре, которая обеспечивает безопасность SDN и использует различные методы, объединенные вместе, чтобы справиться с единственной точкой отказа в архитектуре SDN, добавляет защитный механизм, вводя атакующий источник с ботнетами, использование метода моделирования сетей Петри для определения наилучшего результата предлагаемых топологий и получения различных параметров производительности, которые используются в уравнениях, основанных на этом моделировании этих конкретных топологий, чтобы определить производительность с учетом экономии затрат и сдерживания киберугроз, таких как DoS/DDoS-атаки.

Важной научной решаемой проблемой является разработка нового набора алгоритмов и топологий контроллеров SDN для повышения уровня безопасности SDN и разработка теоретической оценки уровня безопасности компьютерных сетей.

Теоретическая значимость может быть описана путем определения основных проблем в обеспечении безопасности компьютерных сетей, указания основных проблем в парадигме SDN, которые необходимо устранить, теоретизирования основных концепций предлагаемых алгоритмов и топологий.

Практическая ценность работы определяется разработанной структурой, которая представляет собой важный вклад в сообщество SDN, предлагая несколько новых топологий SDN для решения проблемы централизации и защиты соединения между контроллерами SDN. Также обеспечивается лучшая визуализация уровня безопасности конкретной сети путем измерения его с помощью различных математических инструментов, основанных на предлагаемых параметрах.

Внедрение результатов. Полученные результаты были использованы в исследованиях компании Dekart относительно новых подходов к информационной безопасности.

LIST OF ABBREVIATIONS

Acronym	Term
ACL	Access Control List
Ad-hoc	a Latin phrase that means for this purpose
ATM	Automated Teller Machine
ATM	Asynchronous Transfer Mode
API	Application Programming Interface
CDP	Cisco Discovery Protocol
CLI	Command Line Interface
DoS	Denial Of Service Attack
DDoS	Distributed Denial Of Service Attack
DLUX	Open Daylight User Experience
DF	Defense Factor
GETB-SR	GrEen Traffic engineering testbed: segment routing
GETB-AR	GrEen Traffic engineering testbed: Anycast routing
GUI	Graphical User Interface
GSPN	Generalized Stochastic Petri Nets
IoT	Internet of Things
IP	Internet Protocol
IPsec	Internet Protocol Security
IDS/IPS	Intrusion Detection System / Intrusion Prevention System
LAN	Local Area Network
MAN	Metropolitan Area Network
MitM	Man-In-The-Middle Attack
NETCONF	Network Configuration
NFV	Network Functions Virtualization
ODL	Open Daylight
ONAP	Open Networking Automation Platform
OPNFV	Open Platform for Network Functions Virtualization

ovsdb	Open Virtual Switch Database
OS	Operating System
ONOS	Open-Source Network Operation System
OpenLL	Openflow-Based Low-Cost And Low Error Measurement Framework
PoW	Proof Of Work
PoS	Proof of Stake
PSO	Particle Swarm Optimization
QoS	Quality of Service
RF	Risk Factor
RM	Modified Risk assessment
RoS	Reliability of Service
RSA	Rivest–Shamir–Adleman
RESTCONF	Representational State Transfer Configuration
SDN	Software-Defined Network
SC	Software-Defined Network Controller
SDIoT-IDS	Software-Defined- Based Intrusion Detection System For Internet Of Things
SPOF	Single Point of Failure
SHA	Secure Hash Algorithm
TCP	Transmission Control Protocol
TLS/SSL	Transport Layer Security/Secure Sockets Layer
TANR	Total average Number (distribution intensity) of Requests of any of the proposed topologies
$TANR_O$	Total average Number of (distribution intensity) Requests of Ordinary topology
VPN	Virtual Private Network
VLAN	Virtual Local Area Network
VM	Virtual Machine
WAN	Wider Area Network
WSAN	Wireless Sensor And Actuator Network

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

This study aims to increase the security level of the Software-Defined Network by proposing a suite of algorithms and new kinds of SDN controllers' topologies, and elaboration of the new approach to evaluate the security protection of the computer networks, alongside the proposal of four security assessment parameters to measure the reliability level of networks. All that is for the sole purpose of making SDN a safer environment for computer networks that leverage it, which facilitates the transition of classical networks' structure to the SDN structure hence, assuring the security of computer networks in general. Before explaining the need for assuring the security of SDN, it is of great importance to explain the need for SDN in the new era of technology first. Few years ago it was affordable to deal with the data and information when they were still growing using the firstly-created primitive techniques, equipment and classical network management patterns but, it got harder to deal with the growing amounts of data especially with the new era of social media, increment of websites and the emergence of smartphones, tablets, IoT devices etc. that led to create gigantic amounts of data and new ways of managing databases like cloud technologies which in turn required more enhancement to all aspects of computer technology to match that development; except for the essence of the internet connectivity which is the computer network itself since it is the cornerstone of today's communication technology which has remained unchanged and undeveloped since about the 80's and due to the previous reasons it got bigger and harder to manage and maintain especially against the ever-evolving security threats, all that created the necessity to research new methodologies to manage the ever-growing networks' dilemma and one of the promising presented solutions was the technology of Software-defined network SDN. SDN mainly means networks that can be managed programmatically or that can be programmed based on the network administrator's needs. Of course, that will create a whole new horizon of potential opportunities and fields of research to enhance network management, and facilitate policy enforcement in a well-grained, smooth, and swift way, and since it is still a new way of managing networks', it means that even if it solves some major security problems in networks but, it creates new security challenges in the same time. SDN has the great feature of programmability which means that it leverages virtualization which is cost-effective and easier to use in different network infrastructures. In addition, not to forget that despite SDN infrastructure also uses physical hardware, it is mostly software-based which means that SDN will liberate the network users from vendor-constrained products, specs, and standards.

Now going back to the need for this framework, which stems from the diverse ever-evolving continuous cyber threats on computer networks in general and on the SDN in precise, due to the following main reasons, which are:

1. The SDN is still a new field of research, which means that despite their promising capabilities to lead the world of future computer networks but, these networks still need enhancement and due to their novelty, they also raise new cybersecurity challenges.

2. Software-Defined Networks mainly mean networks that can be programmed and that feature itself could be a double-edged sword since that could add more capabilities to the network and frees it from vendor-based constraints. SDN provides network administrators with more tools and abilities to create their own network application based on their enterprise needs and adds more functions to it making the network policy enforcement easier and more robust and agile network management. But, programmability itself could be a hustle since today's widely-spread computer viruses are mostly created using programming. Not to forget, the effect of the human factor on programming since all humans are error-prone and a simple mistake could create a glitch or a malfunction that could be leveraged as a vulnerability by the attacker and that could jeopardize the software-defined networks' structure.

This research provides analysis of security level of SDN. Presents elaboration of a suite of algorithms and technologies incorporated together in one suite to obtain a better security level for SDN in precise and for computer networks in general if they tend to use the SDN structure, since that SDN is not a new network but rather a new way of managing computer networks. In addition, this research suggests three SDN controllers' topologies. It also presents an elaboration of the new method of security assessment and mathematical security evaluation by developing four main mathematical parameters. This study implements the proposed topologies in the three presented topologies resulting a new framework for protecting and managing the SDN structure. This study provides a better understanding of the proposed topologies effect, solely on enhancing and assuring the security of SDN using the Petri Nets approach and make a comparison between them on one hand and compare between them and between the already existing SDN topology which was named in this research as the Ordinary topology on the other hand, to attain four new parameters or equations that could be leveraged to assess the security level of a software-defined network based on the numerical results gained from Petri Nets simulation of those proposed three main topologies.

The importance and relevance of the raised problem is elaboration of a new suite of algorithms and SDN controllers' topologies to increase the security level of SDN and elaboration of the theoretical assessment of computer networks' security level.

The role of SDN security in the industry

- Facilitates network management and policy enforcement.
- Uses a single point of management and control represented by the controller in the control layer.
- That single point of management as it is an advantage it's also a disadvantage since it promotes a single point of failure SPOF.
- The ability to adapt new methodologies and to deal with heritage network structures, tools and devices as well.
- SDN leverages software and/or hardware tools to create a new hierarchy of management.

The SDN security in academia/research.

- New research field that needs more enhancements.
- Dealing with new algorithms as compared to old ones.
- Incorporating already existing other technologies with it and vice versa to enhance it and to enhance those other fields of technology that leverage the SDN paradigm.
- Leveraging new and existing measurement and risk assessment tools (Petri Nets, defense factor equations etc.).
- Increase the response time for networks against some well-known easily-conducted cyber-attacks like denial of service/ distributed denial of service DoS/DDoS attacks especially protocol DDoS attacks.
- Creating a balance between proposed security algorithms effect and their weight on the network traffic speed and interaction.
- Tests in controlled environment (specific topologies effect on promoting and raising the security level).

Developing a model to simulate this perception is still an industrial (by network devices vendors and cyber security society) and an academic challenge. The outcome of this study could be used as benchmark or a framework to be provided and used by the manufacturers to strengthen the defenses of networks and by vendor corporations and network administrators to measure the level of security in a specific network as well.

The main purpose of the research is to increase the security level of the Software-Defined Network by proposing a suite of new algorithms and SDN controllers' topologies for securing data exchanged between multiple nodes.

The research objectives:

1. Analysis of existing methods and technologies for security assurance of computer

networks.

2. Elaboration of security algorithms to provide the SDN paradigm with a better security against cyber-attacks.

3. Elaboration of SDN controllers' topologies for assuring the security of computer networks.

4. Elaboration of the new method of computer networks' security assessment.

5. Evaluation of security efficiency of the proposed SDN controllers' topologies for computer networks security assuring.

The Hypothesis of the research

As the possible solution for the formulated research problems can be the new, more effective algorithms for securing data exchanged between multiple nodes in SDN networks, new kinds of the SDN controllers' topologies for assuring the networks security, and the theoretical basis of the computer networks security level assessment.

The methods used in research

Consist of the methods of cryptography, modeling and simulation, usage of Petri Nets to research the reliability of the proposed approaches of security assurance, security risk assessment law, general economic cost-related data.

The scientific novelty of the obtained results.

Consist of elaborated new algorithms for assuring the security of SDN, which are Hydra, Double RSA, and distributed ledger of Blockchain, the elaboration of new controllers' topologies, which are Serial, Parallel and Hybrid topologies. The elaboration of a new method of security evaluation of SDN technology based on Petri Nets and five parameters which are, Reliability of Service (*RoS*), Defense Factor (*DF*), Risk Factor (*RF*), the Modified Risk assessment law (*RM*) and the Cost effect (*Y*).

Theoretical significance consists of the elaboration of the new basis to increase the SDN security level and proposing a new approach for security level assessment of computer networks.

Results approval

The idea of the research and its results were published in 13 scientific papers, among which 8 by a single author, with a total volume over 7 sheets of author, including, 8 in journals category B+, 4 in journal category B, 1 in journal category C were reported in 5 international conferences and 1 in national (home) conferences (see author's publications on thesis subject).
Both theoretical and practical works related to the subject have been studied to analyze the effect of incorporating SDN with other fields of technology and vice versa, to study the pros and cons

of that incorporation of technologies and to figure out the gaps and weakness points then, patch them up with this research results. This work presents theoretical solutions represented by the proposed algorithms and topologies and some practical comparative analytical results represented by implementing the proposed theoretical topologies using the evaluation procedures of Petri Nets, by leveraging some Petri Nets software tools like PIPE. Those results will be used to patch up and solve the objectives listed in the thesis to provide and assure the security of the SDN environment.

Summary of the doctoral thesis sections

In this study, a full suite of algorithms, methodologies and topologies have been proposed as a security framework to enhance the security level of computer networks in general, by assuring the security of software-defined networks in precise. A simulation of the proposed topologies has been presented to provide a security assessment or a defense factor equation to measure the security of specific software-defined networks' structures and to create a correlation between the proposed topologies and their effects on enhancing the security level of the SDN environment.

The thesis is presented in three chapters.

Chapter 1 includes the motivation, goals, questions, network-related technologies and methodologies of this research. Also, this chapter provides a definition and a historical glimpse for networks in general, a quick look to some of the main network-related nowadays attack and defense techniques, an overview of Software-defined networks' architecture as a potential futuristic consistent and prevailing solution to manage computer networks and facilitate the administration process. This chapter provides an overview and comparison between the classical heritage networks' structure and the SDN structure. A simple methodology of the SDN research lab designing is provided as well. Then, comes a brief clarification of the software-defined networks methodology effect on the market and industry of technology from an economic point of view and how it could be of a big interest to both the leading networking technology manufacturers and to the end users and network administrators as well. Also, adds a general analysis of various methods related to the field of SDN and existing research methodologies regarding objective and subjective experiments; emphasizing the importance of SDN as a potential solution by giving a brief listing and explanation for those technologies correlated with SDN; which were either incorporated with SDN to enhance it or leveraged SDN to develop themselves then, provides a simple review for the pros and cons of each one of them and afterwards, comes a comparison between them in terms of positives and negatives based on their effect on SDN or how they were affected by SDN.

Chapter 2 after enlisting the attack and defense techniques in chapter 1; chapter 2 reaches the SDN solution after that provides the algorithms, methodologies, techniques and technologies which are proposed as a potential solution to enhance and assure the security of SDN environment to make it a safer and a more robust environment and to overcome the obstacles and security challenges in its way to facilitate the transition of computer networks management from classical structure to the SDN paradigm. Those algorithms mainly address and target the two main issues of the SDN structure identified in this research which are the single point of failure and the connection tunnel between multiple controllers which is called the east-westbound API; every algorithm will be discussed in this chapter thoroughly with its theoretical and practical effects where practical effect will encompass the effect of the specified algorithm on every proposed controller topology then, comes an explanation and depiction of the proposed SDN controllers' topologies solution to overcome the centralization issue where the controllers interact in a specific way in every topology and after that comes a simple comparison of the effect of the aforementioned algorithms in terms of their effect on each topology. Also, not to forget that the topologies proposed here may and may not work with the suggested framework algorithms; since that the network administrator may use the framework alongside any of the topologies or may deactivate the framework solution and leave it not unincorporated.

Chapter 3 this chapter provides an explanation of the new method proposal for security evaluation of Software-Defined Networks paradigm. This chapter explains the correlations between the proposed topologies and their effect on security by modeling them using Petri Nets methodology. Based on that relationship between the results acquired, the topologies modelled and the different simulations conducted, it extracts and develops four main mathematical security factors. Those parameters are Reliability of Service (*RoS*), Defense Factor (*DF*), Risk Factor (*RF*), the Modified Risk assessment law (*RM*) and the Cost effect (*Y*) of the proposed topologies. They are could be leveraged to assess the security level of each of the proposed topologies. These factors are formulated basically for the specific needs and requirements of software-defined networks that are based on any of the three proposed topologies. Those proposed five security risk assessment parameters are used afterwards, to model the efficiency of each topology. After that, comes the conclusions of chapter three then, comes general conclusions for the whole thesis.

In conclusion this study, shows the importance of SDN in different fields and the importance of developing some aspects in SDN paradigm to overcome the new ever-evolving challenges. SDN is a productive and easy to use environment with backwards compatibility which means, that it is cost effective as well, with no need to leave the legacy hardware.

SDN gives the network administrators the freedom of policy enforcement combined with agility and results with good quality. internet is coherently related to different fields in today's life and securing it through SDN is one of the prominent solutions, for instance a small cyber-attack on bank accounts could jeopardize a whole nation.

SDN helps control bigger networks in an easy way, which makes it harder for cyber-attackers to infiltrate such well monitored networks.

This study provided algorithms to patch specific issues, like the Man In the Middle (MITM) and Denial of service/ Distributed Denial of Service (DoS/DDoS) attacks, especially those attacks that target the Transport Layer Security (TLS). The study also provided specific controllers' topologies and compared them to the already existing topologies to show the enhancement done by suggesting these topologies. The modelling conducted on these topologies took into consideration the probability of a cyber-attack and the results were gained based on the simulation done. This research also proposed new mathematical equations based on the simulation's gained numerical results. Those equations or relationships can be used as instruments of measurement of SDN performance and its security level against different threats.

The gained mathematical results are mostly theoretical and it is well considered that division over zero is not permissible but due to the results gained by the PIPE software simulation; it was imperative to conduct such a mathematical operation, but this shows that the Parallel topology is very reliable and near optimal.

1 ANALYSIS OF EXISTING METHODS AND TECHNOLOGIES FOR SECURITY ASSURANCE OF COMPUTER NETWORKS

Networks hierarchy and structure have remained almost unchanged since decades and with the emergence of Software-Defined Networks paradigm, it was obvious that it could be the solution for many classical network structure problems like the difficulty of managing enterprise networks containing hundreds or thousands of switches and other network nodes, policy enforcement, the cost of deploying classical networks' hardware technologies. But, with this new solution represented by the SDN comes other problems and new security challenges especially that it is still almost a new technology so, its potential threats are still not explored to the peak and that means solutions for some issues and having new security issues. This chapter gives an overview and description for the computer networks in general then SDN afterwards and tries to list some researches related to developing SDN or leveraged SDN to develop other fields to show the importance of SDN technology and also to analyze the main strength points to consolidate them and analyze vulnerabilities in the SDN structure to help patch them up. The main goal of this research is to design a new framework that could be leveraged by the SDN paradigm to overcome some of the issues existing in the SDN structure and to formulate an SDN-based security level assessment equation; and all that to facilitate the transition of managing the computer networks from classical paradigm to the SDN paradigm.

1.1 General Overview Regarding SDN Structure, Cyber Attacks And Defense Techniques, Networks And Information Security

First this thesis gives a definition of the network, its structure, and its importance, then talks about the Software-Defined Network, its structure as compared, how it can be designed and established, then comes listing of some defense and attack techniques. A network consists of a computer or computers and some other devices and terminals that are all connected together by a device called a switch and to connect more than a network with each other it is possible to use a device called the router, now when talking about the flaws of the infrastructure of the old network, it is possible to see that it depends on a principle called distributed management. As shown in the Figure 1.1.

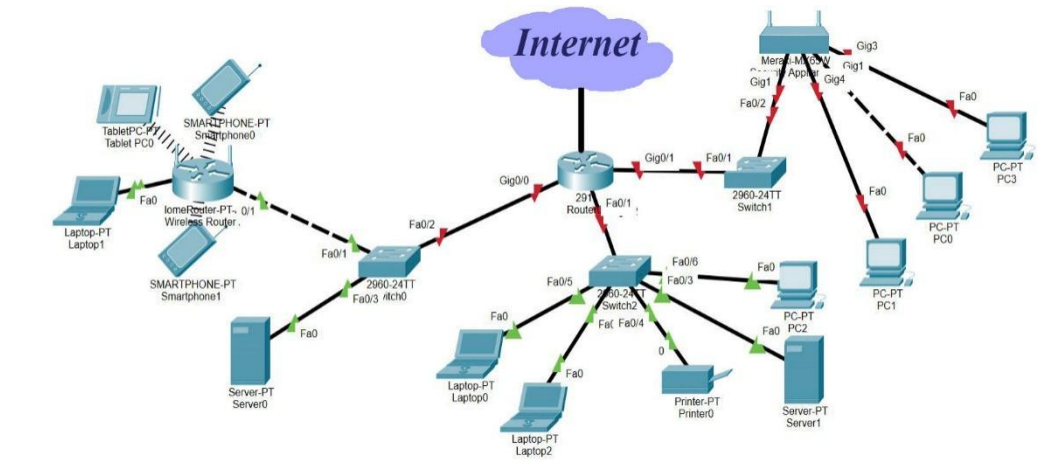


Figure 1.1. A Network Example

This implies that all security configurations and the rest of settings have to be set up manually for each single device, which will be a tedious procedure to do; especially with large scale networks like enterprise networks that could contain more than 1500 switches so, as known that classic structure of a network contains a seven OSI (open system interconnection) layers which are as shown in the Figure 1.2.

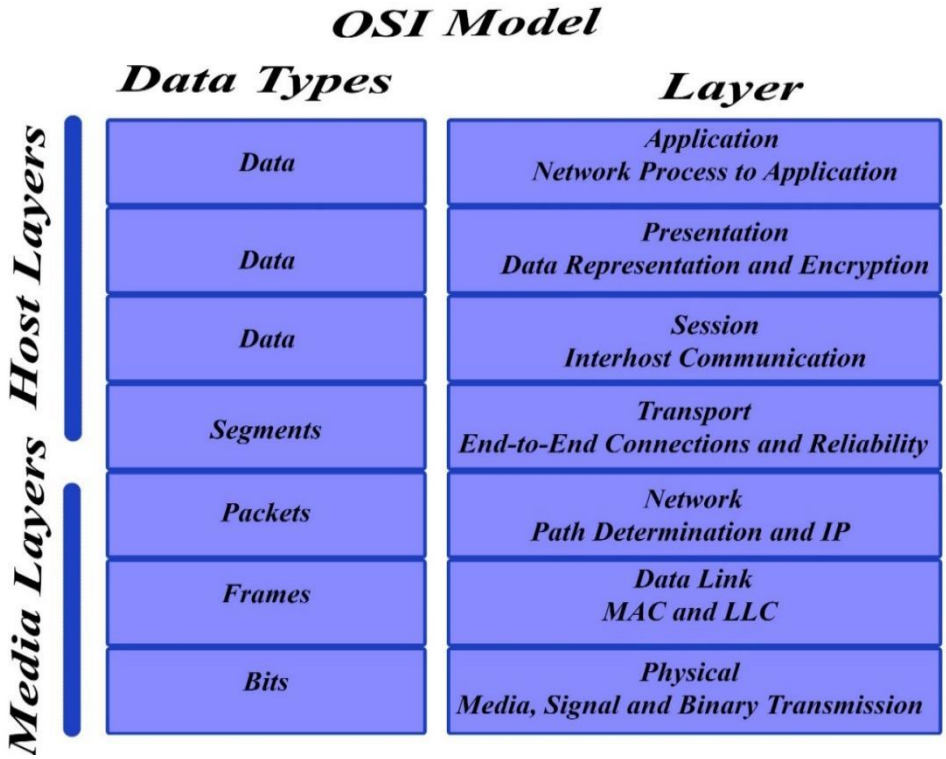


Figure 1.2. The Open System Interconnection (OSI) networking model

Since every computer or device in the modern world, is connected in a way or another to a network whether it was a local area network (LAN) or a wide area network (WAN) and these host devices (computers, Ipads etc.) contain delicate people's information then it is needed to

provide a decent security level for the networks they are connected to, Due to many reasons especially:

- Hackers and viral attacks: some people predict that the wars of the future will not be even nuclear, imagine if a 20 years old guy can manipulate a nuclear arsenal that is located thousands of miles away from him, especially with the last viral attacks with ransomware and Petya malware there is a bigger motivation for enhancing network security.
- Privacy violation: people's life is not just about their money, there could be delicate, sensitive information about their lives that could worth more than money to them and accessing that information to blackmail them is against all laws and ethics worldwide.
- National Security: as mentioned before, waging a war against a country doesn't necessarily has to be by missiles or tanks, the world is entering a new era of wars, protecting a country's networks and internet from viral attacks is a protection for its national security, its people's lives, automated gas pipelines, traffic, T.V channels live stream, stock market etc.
- Network speed disruption: finally, it is needed to know that the internet itself was designed and made as a connection means for the military in the first place, and now after it entered the civil service to be available for everyone, it is imperative not to forget that disabling the internet network itself for a certain country or slowing its speed is a gigantic problem for that it is of a great deal of importance for daily use since it has turned the whole world into a small village and controls different life aspects like trains, flights, vehicles traffic etc. There are some flaws in the infrastructure of this network, since that it depends on a principle called distributed management, so a researcher named Martin Casado suggested a solution named Ethane, which became the new design for Software Defined Networks which depends on the separation of control plane and data plane in network devices, hence giving control to one device whether it was a hardware (dedicated controller) or a software (virtual machine program), and let all other devices just do packet forwarding. It is worth mentioning that the OSI or the model that came after it which is TCP/IP; still exist within the new structure of network management (SDN paradigm) but this topic is for the purpose of showing the agility of SDN environment, because for instance the openflow protocol works through the transport layer security TLS or transport layer.

The layers of the new management structure of SDN technology are:

- Management plane.
- Control plane.
- Data plane.

It is known that the 3 main elements in all network devices routers/ switches which are: control plane /data plane respectively and the data itself, where the control plane is the mind (the

decision maker) meaning that it controls how, where and when data is being sent, regarding data plane, it is like the (muscle), it executes the decisions of the control plane on the data to transfer them between two points, that's why it's also called forwarding plane, data is the information being exchanged between different network devices. They found out that the switches are not opened or allowed to take orders from a higher level (control plane) and that would mean that the infrastructure of the network would remain constrained due to the structure of the network switches, so researchers proposed a new protocol called OpenFlow protocol which is responsible about creating a communication channel between the control plane that is represented by the (controller/controllers) and the forwarding plane and that is represented by the switch. They organized all the components affiliated to the forwarding procedure and put it into a frame called Flow Tables, taking into consideration all the common information of different vendors. The figure 1.3 depicts the general structure of SDN.

Methodology of the lab designing.

There are different software tools used to create SDN environment on which it's allowed to test new ideas and methods on, one of them could be mininet which is an SDN simulation software and also Graphical Network Simulator-3 (GNS3) could be used for that purpose by installing mininet within it.

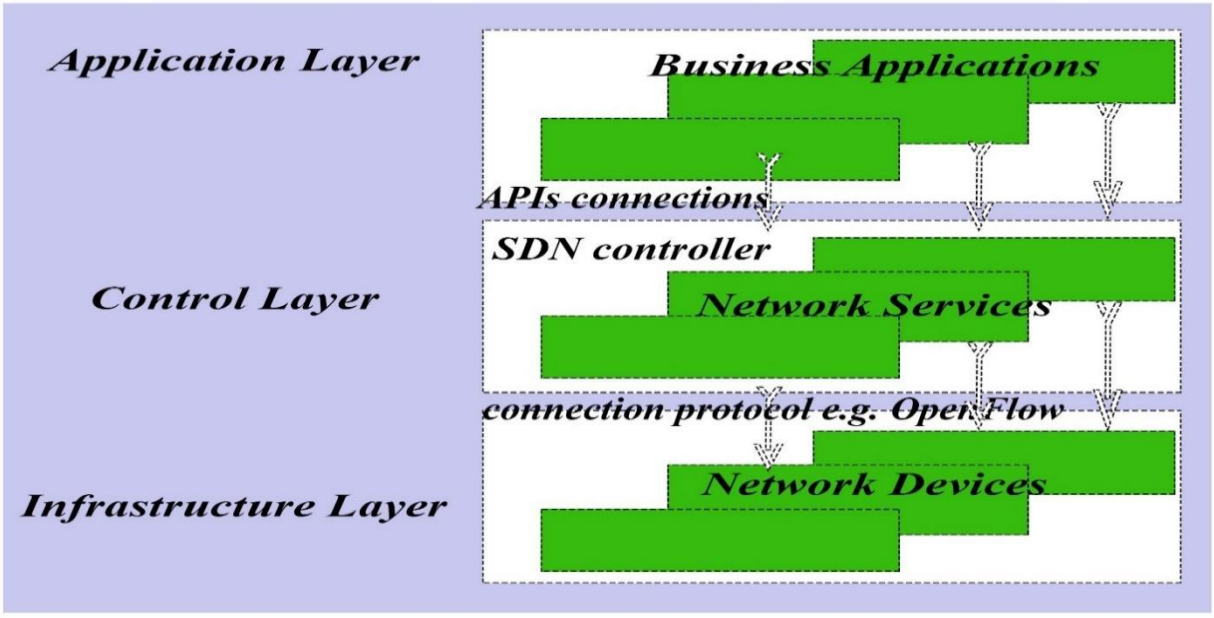


Figure 1.3. The Structure of the New SDN Model

After testing the network and make it work, simple ways will be used to measure the efficiency of the network, there are many software types for that purpose, for example:

- Cbench: an openflow controller benchmarker, it's a special tool used to measure the

efficiency of the OpenFlow controller by creating a set of various switches that send messages of type packet-in to the controller and monitors the response and based on the reading it measures the performance and ability of the controller [1].

- OFLOPS: An OpenFlow switch benchmarker, it's a tool that does the opposite role of the previously mentioned tool, it measures the efficiency of the OpenFlow switches by creating virtual controller that sends messages to the switches and monitors their responses and measures the performance of the switches based on that reading [1].
- OFtest: it's a tool used to test the embedded OpenFlow protocol in the switches, it supports up until the 1.2 version of that protocol [1]. The Figure 1.4 below shows a simple example of a software-defined network environment.

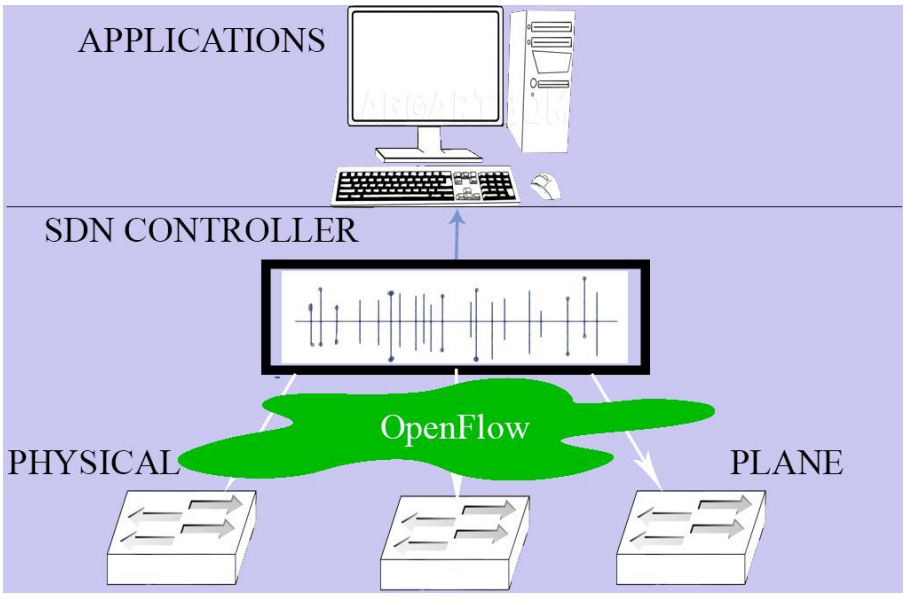


Figure 1.4. An example of a Software-Defined Network

General Cyber-Security Attack and Defense Techniques.

Many viruses and network and computer disruption methods emerged in the last 3 decades; ransomware and Petya malware attacks are new and a bigger motivation for enhancing network security. Also, the virus attack called (Stuxnet) in 2010, that compromised the centrifuges in Iran's nuclear reactors was able to disrupt them as well and this is another evident example of cyber security's effect on a state level [2]. The first hearing in congress for technical experts in cyber penetration (known as hackers) took place in the 90's [3], and that helped giving awareness to both the government and general public about cyber world issues and its vulnerabilities. Today the cyber-space or internet is everywhere and an example of internet's importance nowadays is the internet blocking in Iraq back in July 2018 for few days to prevent

the usage of social media websites by protesters, which costed Iraq about (40 million dollars a day) [4].

Attack Techniques.

Now talking about some security threats by giving a simple description of some of the main types of attacks, and here are some of them:

- Denial of service (DoS) and distributed denial of service (DDoS) attacks: DoS attack can be described as a cyber-attack in which the attacker tries to stop or disrupt the resources of a network or a machine whether software or hardware and deprive their intended users from using them. While the DDoS occurs when the attacker uses multiple system to flood the resources of the target to stop it from serving its users and of course DDoS is much stronger than DoS and it's conducted using botnets to control multiple hosts, where botnets are a group of malicious tools acting as an entity. Furthermore, history has it that the aim of what gave rise to botnet was the idea to simplify the method of message exchange within networking platform [5]. Another way to describe DoS/DDoS attacks is that they are typically explicit attempts to exhaust victim's bandwidth or disrupt legitimate users' access to services [6].
- Man-in-the-middle (MitM) attack: Which is an attack done by attacker secretly dwelling and maybe manipulating the communication between two parties without their knowledge or as it is described in [7] is an attack where the attacker furtively transfers and perhaps changes the correspondence between two parties who trust they are straightforwardly communicating with each other.
- Phishing and spear phishing attacks: Here the attacker tries to disguise as a trustworthy entity or party in the communication mean to gain sensitive data which is he unauthorized to get such as login credentials. And when this attack is aimed at specific targets whether individuals, organizations or companies then it's called spear phishing. Phishing is known also as the social engineering since it uses some social means like emails or social media pages [8].
- Drive-by-attack: An unauthorized unwanted program that is downloaded with authorized wanted program by the user permission without his knowledge that it contains malicious code or files, also it is called the drive-by-download which refers to attacks that automatically download malwares to user's computer without his knowledge or consent [9].
- Password attack: Also called password cracking, in this type of attack, the attacker tries to gain or recover the password data being stored or transmitted; one of the most famous methods used is Brute-force attack. The typical way password cracking works is to get a file containing user hashed passwords and then run a cracker against the file to try to get matches for all of the hashes, thus revealing all of the passwords in the file [10].

- SQL injection attack: is a method of exploiting the database of web application. It is done by injecting the SQL statements as an input string to gain an unauthorized access to a database. SQL injection is a serious vulnerability that leads to a high level of compromise - usually the ability to run any database query [11].
- Cross-site scripting (XSS) attack: Here the perpetrator uses some vulnerabilities which exist in websites and tries to inject some client-side scripts into the webpages of the targeted site [12] or it could be defined as stated in [13] which is that (XSS) is a completely, a generally exploited vulnerability which could be very extensively unfold and easily detachable.
- Eavesdropping attack: This attack is considered as a passive attack since the perpetrator dwells and awaits while captures fragments of information and some packets and forms them trying to get any piece of information, usually is conducted on wireless networks [14].
- Birthday attack: In math there's probability theory topic and in that topic there's a mathematical problem called birthday problem, the attacker tries to exploit this problem and it could be used on cryptography applications, the birthday attack makes use of what's known as the Birthday paradox to try to attack cryptographic hash functions Among other desirable properties of hash functions, an interesting one is that it should be collision-resistant, that is it should be difficult to find two messages with the same hash value. To find a collision the birthday attack is used, which shows that attacker may not need to examine too many messages before he finds a collision [15].
- Malware attack: Malware is derived from malicious software, which is any malicious code made to cause harm or exploit a vulnerability in a system or a device another definition could be used as in [16] which states that it is a program code that is hostile and often used to corrupt or misuse a system.
- Data leaks: It is described as the unintentional or intentional exposing or releasing of delicate information to unsecure or undesired environment. Personal data breaches from organizations, enabling mass identity fraud, constitute an extreme risk [17].
- IoT vulnerabilities: Which could be used or exploited by malicious entities to disrupt the whole system of IoT connected to the internet; some of them are insecure passwords, usage of outdated devices etc. [18].
- Timing channels: It is an attack used to change the timing of events in the execution process. From all the previously mentioned, it is seen the importance of elaborating a highly protected computer networks and assure their security
- In an attempt to match the current and future needs and reach the level of ideally

secure computer networks to create a secure cyber world that can be a safe environment for individuals, entities and countries' governments. There are many surveys and studies conducted on that matter [19].

Defense Techniques.

After briefing about some cyber-attacks, now is the time to mention some of the main old and new techniques and methods used for cyber-space security:

- VPN: It has various technologies [20] in securing the communication between networks.
- VLAN: The development of virtual local area network VLAN led the great organizations such as companies, universities, enterprises etc., to build their network schemes depending on VLAN encapsulation method [21].
- Packets Filtering: In this paper [22], the researchers propose and introduce an analytical dynamic multilevel early packet filtering mechanism for firewall performance enhancement.
- Packets dropping: it is used to drop the packets that don't match the flow table rules and it could be as an attack in some cases and it has some tools to detect it [23].
- Traffic quarantine.
- MAC and/or IP address change.
- Intrusion detection system/intrusion prevention system an intrusion detection system (IDS) is software that automates the intrusion detection process. An intrusion prevention system (IPS) is software that has all the capabilities of IDS and can also attempt to stop possible incidents IDS/IPS [24].
- Deep packet inspection: Deep packet inspection (DPI) helps Internet service providers in efforts to profile networked applications. [25].
- Reconfiguring the network and changing the topology.
- Bandwidth control: in [26] the researchers give an example of the importance of the bandwidth in networks by proposing a new network design using top-down methodology which takes into consideration the business and technical goals of the corporate. So, in order to improve the bandwidth utilization efficiency, they evaluate few traffic shaping methods in terms of bandwidth utilization, latency, and packet loss.
- ACL. Extranet is a popular network among most of the organizations, where network access is provided to a selected group of outliers. Limiting access to an extranet can be carried out using Access Control Lists (ACLs) method [27] which will provide the Extranet with a better security level.

- SDN, which is the software-defined networks which this thesis discusses.

As understood from above, there are different technologies for cyber-defense and they differ in the importance, goal, and the way it's implemented but, SDN could be a new era of managing the network technology since it's able to combine many technologies, also it opens the horizons for further development of services and ideas that could assure the security of computer networks.

1.2 Software Defined Networks And Their Effect

Due to the previous security reasons mentioned before and also due to the increasing data and soaring internet speed, the ethane design was suggested and its structure layers are as follows:

- **Management plane layer** the management plane contains the graphical user interfaces (GUI) and services needed to manage the network and has the traffic used by the network administrator to manage the network. The traffic is characterized like this:
  - Management plane: represented by user devices to configure devices (like SSH).
  - Control plane: a device to determine actual forwarding policies (e.g., OpenFlow or BGP (border gateway protocol) updates).
  - Data plane: the traffic that is forwarded through the network (e.g., Hyper Text Transfer Protocol (HTTP) sessions, VoIP etc.).
- **Control plane layer** or control loop and that is the SDN brain containing the controller (Whether it was a hardware/software-based) that manages data plane's operations using southbound application programming interfaces (APIs) like openflow protocol which is a Forwarding and Control Element Separation (ForCES) created by the Internet Engineering Task Force (IETF). There is a virtualization technique for solving the traditional problem but when considering server virtualization, it is limited to 4000 VLANs in layer two. Open networking foundation (ONF) which is a non-profitable [28] organization has created OpenFlow, which contains multiple feedbacks to increase the network accuracy. OpenFlow contains some advantages like controlling packets rate through flow basics and also each packet has cookies added to it. It's needed to note that management plane is sometimes considered as a subset of the control plane [29].
- **Data plane layer** which is the infrastructure layer where it contains the data packets forwarding devices (switches) which are relieved of any sophisticated responsibilities except for forwarding data and managing their routes.

There has been already a whole arsenal of methods and mechanisms created to assure the safety, confidentiality, integrity and availability (CIA) of data transmitted over the network channels. As in the Figure 1.5 which shows the SDN structure layers with the services they provide [29].

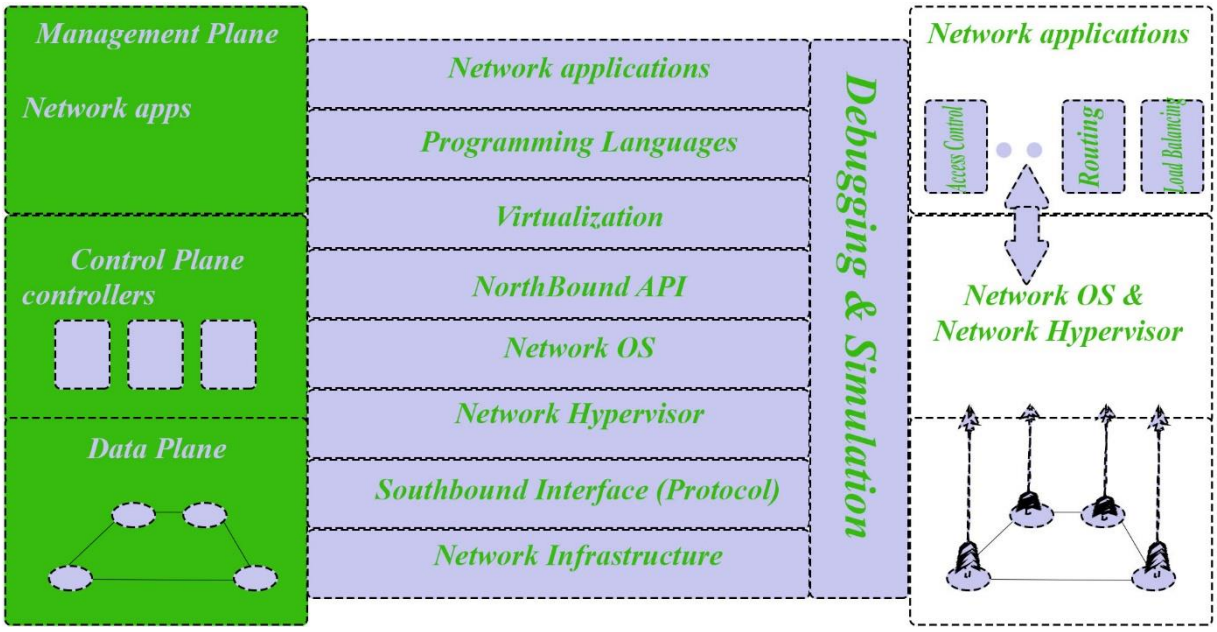


Figure 1.5. Software-Defined Network structure's layers and their services

Traditional Network VS Software-Defined Network.

There are two factors that recognize the main differences between traditional network configurations versus SDN configuration:

1. Network functionality is mainly implemented in a dedicated appliance or device. In this case, (dedicated appliance) means one or multiple switches, routers and/or application delivery controllers.

2. Most functionality within this apparatus is implemented in dedicated hardware. ASIC (Application specific integrated circuit) will be used for this purpose. Organizations are increasingly confronted with the Limitations that accompany this hardware-centric approach, such as:

- Traditional configuration is time-consuming and error-prone: Many steps are needed when an IT administrator needs to add or remove a single device in a traditional network [30]. First, here will have to manually configure multiple devices (switches, routers, firewalls) on a device-by-device basis [30]. The next step is using device-level management tools to update numerous configuration settings, such as ACLs, VLANs and Quality of Service [30]. This configuration approach makes it that much more complex for an administrator to deploy a consistent set of policies. Hence, organizations mostly will encounter some security breaches and

non-compliance with implications. So, in other words the highly administrative obstacle is that, the traditional configuration interferes with meeting the standards of business networking.

- Multi-vendor environments require a high level of expertise: the average organization has various sets of equipment of different manufacturers. The network admin will need to acquire comprehensive knowledge of all device types and vendors in order to successfully configure and set the network [30].

- Traditional architectures complicate network segmentation: a development further complicating networking matters, is the connectivity evolution that is currently occurring. In addition to tablets, PCs and smart phones, other devices such as alarm systems and security cameras will soon be linked to the internet. While SDN offers network virtualization and that is creating virtual networks separated from physical network parts. The combination of the global or network-wide view and the network programmability supports a process of harvesting intelligence from existing Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS), for example, followed by analysis and centralized reprogramming of the network. This approach can render the SDN more robust to malicious attack than traditional networks [31].

SDN Effect.

- Cost saving: when building and managing the infrastructure devices for the used network; there will be no need to change the whole infrastructure. Also, virtualization plays a major role in cost saving.

- Programmability: that provides better accuracy and flexibility in the design, application, management, security improvement, time, speed, automation, reliability, problem solution and error overriding, which suits largely the big needs and future continuously changeable services [30].

- Optimizing Data Flow: A second expected business benefit of the SDN approach, is the optimization of data flows. So instead of having one path from the communication flow source to its destination, an SDN controller is capable of specifying multiple paths per flow. Also, this method permits to split the flow's traffic across multiple nodes [30].

SDN Security Effect.

The new structure of the SDN paradigm brings great benefits to the networking scheme. The way the functionalities are abstracted in the SDN paradigm allows writing high-level software applications to manage the network without worrying how to configure the underlying physical network [32]. Some of the SDN effects in securing networks are the following points:

1. Centralized Data Routing: One of the main features of SDN is its capability to guide

and route all traffic through a one single centralized controller used to route packets through a single firewall which will increase the efficiency of IDS and IPS data capture.

2. Providing a More Efficient Policy Management: instead of the need to physically configure security solutions, SDN leverages central control and administration for the whole security policies management.

SDN industrial effect in the market of technology.

The sector of networks generally is a thriving sector since the dawn of technology history, even the market of classical networks is still soaring up, now with the developing trend taking place in the networks world, it will get even higher attention from general public. This is the new era of technology, it's not transistors, intercontinental missiles but it is networking. The Figure 1.6 shows the market size of software-defined networking.

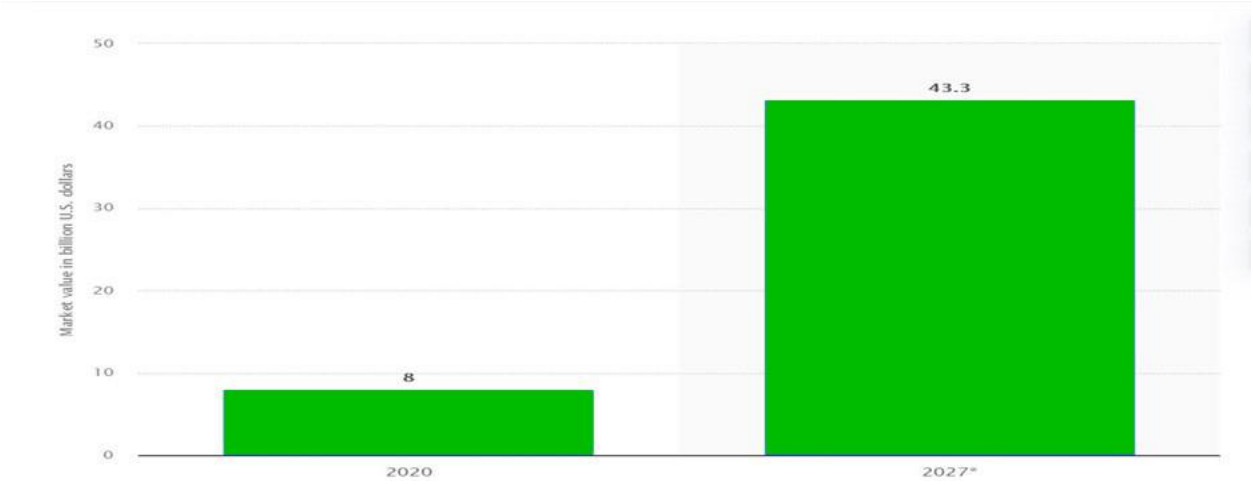


Figure 1.6. Software-defined networking (SDN) market size worldwide from 2017 to 2021 (in billion U.S. dollars) [33]

The statistics shown in the figure above; depict the size of the software-defined networking (SDN) market worldwide from 2020 to 2027. In 2020, the global software-defined networking (SDN) market reached 8 billion U.S. dollars in size. In the United States, the market is estimated at 4.1 billion U.S. dollars that year. Geographical regions within this market projected to experience growth are China, Japan, Canada, and Germany [33].

1.3 General SDN-Related Techniques

The following works show the importance of SDN since they incorporated other technologies to develop SDN or used SDN to develop other technologies:

1.3.1 Network performance measurement

Many attempts were done to develop control and measurement in SDN management due to their significance and mostly more have been done for the control besides, the centralized control feature of the controller makes control process more sophisticated. One of them is this

research to develop a low-cost high-accuracy software-defined measurement design [34]. There are two ways to measure the network performance: active and passive. Active measurement needs to inject probe packets into the network for monitoring their behaviors, such as popular ping and iperf, active measurement may violate the network behavior and produce a negative influence on the measurement accuracy. While passive measurement can cause a little overhead, such as Simple Network Monitoring Protocol (SNMP), NetFlow and sFlow, which are widely used in networks, passive measurement requires a number of network devices [34]. A typical flow monitored by NetFlow consists of three main components: flow exporter, flow collector and analysis application. The sFlow is formal industrial standard protocol used to monitor networks with high speed. Scalable technique, a low-cost solution and providing a network-wide view of usage could be considered as a positive side in the sFlow [34].

Architecture and System Design.

This research presents OpenLL, an adaptive sampling algorithm that reduces network overhead and improves accuracy. Unlike OpenNetMon, OpenLL polls only edge switches (first and last) to measure throughput while adjusting delay for precision. It also introduces a low-cost topology discovery mechanism and leverages multipath for better network topology consideration. As shown in Figure 1.7.

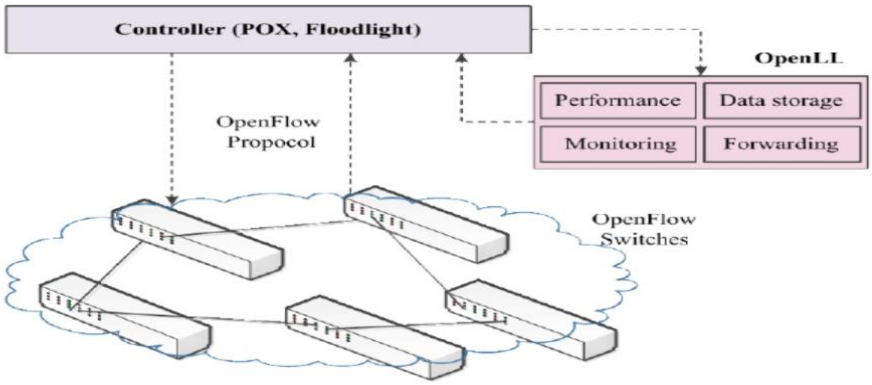


Figure 1.7. OpenFlow-based Low-cost and Low-error measurement architecture [34]

Description.

Ip erf will be used to send packets to the server from the client in order to generate a traffic through the active measurement. Then the controller prompts the first switch to send a probe packet that will traverse the path till the last switch and returns to the controller from the last switch. The leaving time of the packet from the controller to the first switch can be described as *TA*. The time consumed by the packet to arrive from the last switch to arrive to the controller is abbreviated as *TB*. The round-trip time *RTT,* from the controller to the first switch is *RTT1* and from the controller to the last switch is *RTT2*, also there's addition for the standard delay value is

set to $\bar{\iota}$ and of course not to forget the time delay threshold $\xi\ 2$ that will help improving the measurement accuracy, so the time delay in terms of equation in this framework it can be like the following:

$$T\ delay = TB - TA - \xi 1\ (RRT1 + RRT2) + \xi\ 2. \quad (1.1)$$

Where $\xi 1$ is considered to be the proportionality coefficient and it is ranging from 0 to 1 and $\xi\ 2$ from $-\bar{\iota}$ *to* $\bar{\iota}$. Also, to calculate the packet loss by the packet counter:

$$Ploss = (Packet\ f - Packet\ l) / Packet\ f. \quad (1.2)$$

Where *packet f* refers to the packets counter of the first switch and *packet$_l$ is referring to the packets* counter of the last switch. Also, it's possible to define throughput based on this paper as shown here:

$$B_t = 1/2\ (B_f + B_l). \quad (1.3)$$

Where B_f represents the byte counters of the first switch while B_l refers to those of the last switch. The experiment is shown in figure 1.8 below.

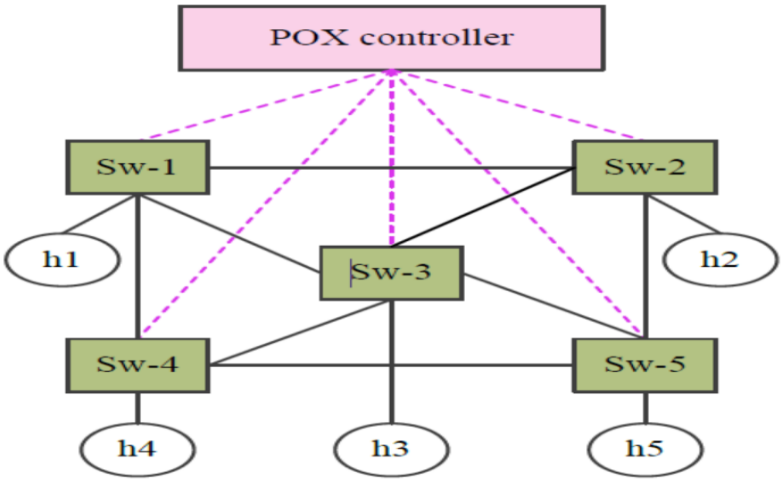


Figure 1.8. Network Topology for experiment [34]

It is possible to define the accuracy of the experiment conducted in this research using the root-mean-square in the following equation:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2}. \quad (1.4)$$

Where X_i, represents the throughput calculated by the measurement architecture, *n*: is the polling times selected by this research's measurement architecture.

Positives and negatives.

Positives: this research is a promising one providing a framework that works alongside with SDN controller and resides within it, and that alone can be an achievement since it doesn't need to be integrated with a cloud or to have any add-ons; another important thing is that it

provides enhancement for the OpenNetMon framework on which this framework is based, and the enhancement is better accuracy and less network overhead.

Negatives: the proposed OpenLL framework which is based on the OpenNetMon framework that uses an active measurement method wouldn't just affect the measurement accuracy but also poses a negative manipulative effect on the network behavior despite that some ways to reduce the overhead and enhance the accuracy were proposed. Also, the link delay threshold doesn't measure the time required or consumed in packet processing within the switch.

Related works.

There are many other works of this type like Open-NetMon [35] which is offering an active approach and open-source software implementation for monitoring per-flow metrics, like packet loss, throughput and delay, PAYLESS [36] which provides a flexible RESTFUL API to help monitoring link utilization that leverages an algorithm for adaptive statistics collection. also, the FlowSense [37] that offers a method for passive monitoring, which leverages the PacketIn and Flow Removed messages to evaluate per-flow link utilization. Although, communication overhead but, it's noticed that FlowSense has comparatively low accuracy. OpenSample [38] suggests an SDN-based sampling measurement algorithm, which uses sFlow packet sampling to provide near real-time measurement accuracy, which is applied on the Floodlight OpenFlow controller. OpenSketch [39] suggests an active software defined traffic measurement design that makes sketches more elastic and flexible to support various measurement tasks with high accuracy and low latency.

1.3.2 Timing channel detection

Covert channels are used for information transmission in a manner that is not intended for communication and is difficult to detect [40]. This work [41] is conducted to detect timing channel (TC) threats. This threat has the purpose of leaking and transmitting information by disrupting the ordering or timing of events. The current TC detection approaches are either anomaly-based or signature-based. Anomaly-based signature uses legitimate traffic modeling to detect unknown timing channels TCs. This work [41] offers a new framework named as OBSERVER that uses elastic resources available in the cloud, it is attached with a new metric that first decomposes the timing data of network flows using the discrete wavelet-based multi-resolution transform (DWMT). Then it implements the method of Kullback-Leibler divergence (KLD) to measure the difference of flow pairs. The OBSERVER structure is shown in the Figure 1.9.

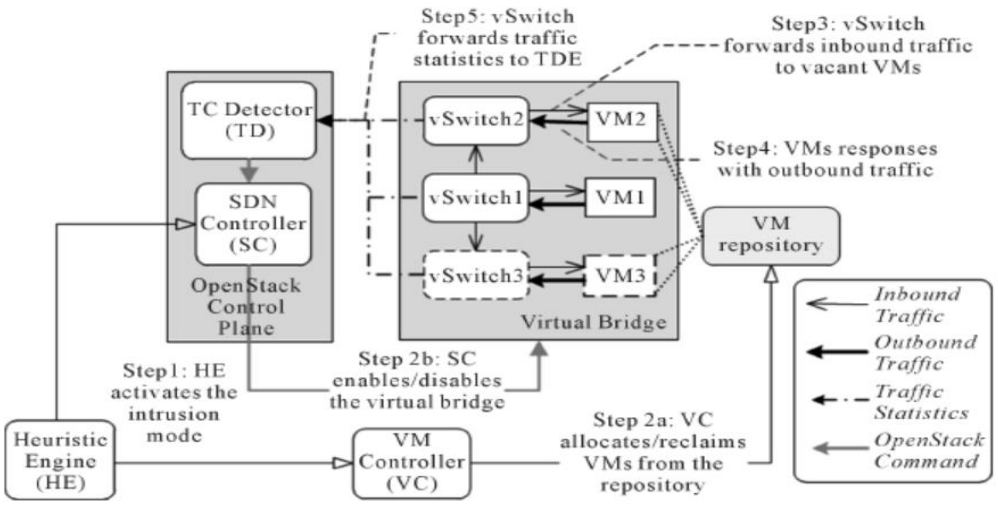


Figure 1.9. System architecture of the OBSERVER [41]

Positives and negatives.

Positives: this research shows a promising ability to detect, analyze and counterattack or mitigate one of the prominent types of threats on the SDN environment itself which is the timing channel attack that can change the timing of events, in other words it provides a better security and enhances protection for SDN especially against many types of TC threats like Covert Timing Channels and Virtual clock noise in Virtual Machines.

Negatives: despite its big success rate but this design mainly depends on connecting the SDN environment to the cloud and as known the cloud environment is controversial itself since it has some privacy concerns like the ability of the service provider to access the data resident in the cloud at any time. Also, according to the cloud security alliance [42] report which stated the most prominent threats posed on the cloud environment and some of them are:

- Insecure interfaces and APIs.

- Denial of service.

So, the cloud environment itself is under threat, and it’s not feasible to depend on it 100% as the researches did here, because for example they didn't suggest adding the TC detector or heuristic algorithm to the controller but they used what's available in the cloud instead.

Timing channel Detection-Related Work.

There have been many attempts and researches that are based on the TCs detection such as Berk's research [43] that leveraged a method of simple binary covert timing channel which is based on the Arimoto-Blahut algorithm, which computes the distribution of input which maximizes the channel capacity, also Cabuk [44] offered a method for detecting the IP timing channel (IPTC) threat and the time-replay timing channel (TRTC), while Shah [45] introduced a keyboard device that could leak information typed over the internet slowly and named this

device as JitterBug. Giffin [46] proved that even if it wasn't a TC but the TCP timestamp low-order bits could be used to create TCs due to the sharing of packet timing and statistical properties of timestamps.

1.3.3 Deploying and managing SDN as a home network

Despite that one of the biggest motivations for SDN technology was the big amounts of data and large data centers but still SDN can be leveraged as a management way for home networks, that's why this survey [47] shows ideas about the SDN deployment in home networks.

Positives and negatives.

Positives: it highlights the importance of software-defined network technology and its vital role in different aspects of the cyber world, giving a full survey and creating a small database-like for researches and works related to SDN deployment in home environment.

Negatives: despite that the research was good but it didn't explain or give a brief of the algorithms mentioned in the works stated in this research, for example, what algorithm and how does it work or a simple sketch for it describing its work.

Related works.

This field of development also had its own share of development attempts, so for starters it’s possible to take, the method introduced by the work that was a part of the homework project of The University of Nottingham, 2012 [48] which tries to study or give a general perspective of using SDN technology in home networking, through redesigning exiting home-network infrastructure (such as routers) based on the concepts of SDN to provide the user with better understanding and control as well as novel interfaces [49], [50]. The works are categorized into the following:

1. General topics, the engineers in [51] try to study the case of the home network to discuss the utilization of SDN to refactor current networks and give users the right level of network actionable information and visibility. While, it’s possible to see that the concept of virtualization is suggested in the rest of the works of this generic category [52], [53].
2. Some of the prominent researches related to this category is the quality of service (QoS) and the quality of user experience (QoE) [54], where the general purpose of these articles is to develop the multimedia and video streaming and make a better bandwidth allocation for various network applications to develop the user experience. The optimization is user preferences-based or profile-based mostly, but it can be also derived from collected-traffic-statistics-based dynamic traffic shaping [54], or automatic identification of applications [55] or it could be like a suggested bandwidth allocation algorithm like what this article [56] suggested. Another idea depends on local solution using in-home SDN controller to control the QoS [54],

another research suggests a novel pricing model for ISPs that can execute a time-dependent hybrid pricing scheme through the APIs of SDN [57]. There are many novel researches mentioned in this survey [47] that can form a great establishment for the topic of implementing SDN in home networks, like the work of [58], which addresses the main issue related to the usage of IoT devices in terms of smart homes, other works were introduced to address Internet use management through the SDN architecture [59]. Works like [60] which designs a system and proposes a method to collect the statistics of usage and sends them back to the controller that information for displaying then deciding and enforcing the proper policies.

1.3.4 Securing IoT with SDN-based IDS

Here the work [61] that introduces an approach for securing the internet of things infrastructure through the usage of intrusion detection system that is based on the SDN approach. The internet of things term is getting more familiar day by day; it adds a revolutionary push for the turning of ordinary devices to smart ones. But this technology is still vulnerable for various threats so, here comes the SDN as a defense mechanism. In this research [61] the researchers used the intrusion detection system based on the software-defined network technology and turned it into a countermeasure against threats that could perturb the IoT network.

Main threats on the IoT environment.

It is difficult to provide a complete security for all IoT devices due to the limited resources, that's why cyber-attacks are soaring up, threats on IoT environment can be categorized into:

1. Botnets: Bot as it is popularly called is an inherent attributes of botnet tool [5].
2. Distributed Denial of Service (DDoS) attacks, which ban the user form using resources

or make resources unavailable to user by keeping network busy with useless and meaningless traffic.

3. Malware or Virus: Malicious code that infects IoT devices, causing damage like data theft or unauthorized access.
4. Cyber-theft or Data Theft: critical, crucial and confidential information could be

retrieved from the network due to the poorly protected IoT devices.

Software Defined IoT-IDS.

The system proposed in this research, is embedded into the controller of SDN since the SDN controller can be used for easier policies enforcement, besides that the SDN controller provides better overall control and management for the network's activities and applications, the researchers here use the SDN gateway for routing traffic in the IoT network. The new messages

are redirected to the controller using the SDN gateway in order to help the controller generate the proper flow rules. The researchers depicted the whole mechanism as shown in the Figure 1.10.

Software Defined IoT-IDS is formed by the main following components.

- **Activity monitor,** which performs the task of monitoring the traffic flowing in the IoT environment.
- **Activity analyzer** is used to detect a specific type of a network attack for that it has a machine learning algorithm that helps it in this task. Another element is added to this important part of the system, which is Back propagation neural network (BPNN) to enhance detection purposes.

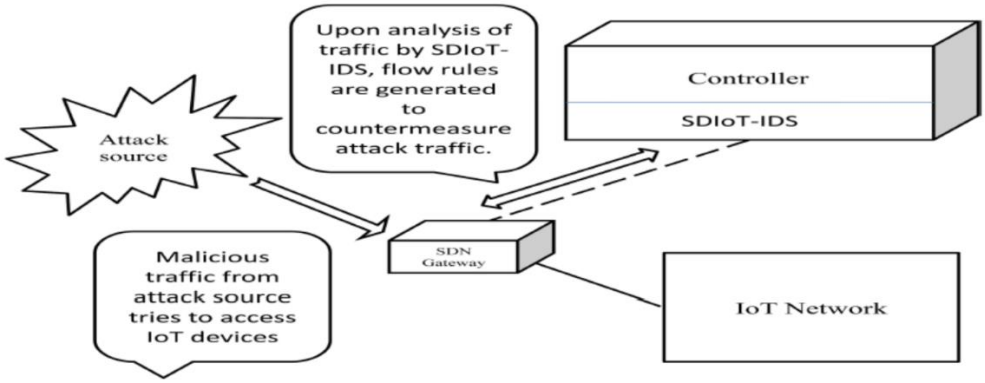


Figure 1.10. Working of SDIoT-IDS [61]

The nodes of the input and hidden layers of the BPNN are connected to the nodes of the output layer and each node of the hidden player is directed to one output, then the output O will be calculated as in the following equation:

$$O = f(a1 \cdot w1 + a2 \cdot w2 + a3 \cdot w3). \quad (1.5)$$

In the figure 1.11 below; the researchers show how the nodes are connected in feed-forward method in BPNN. Where $a1$, $a2$, $a3$ refer to inputs at various levels, $w1$, $w2$, $w3$ represent the weights and $f(\bullet)$ refers to the activation function.

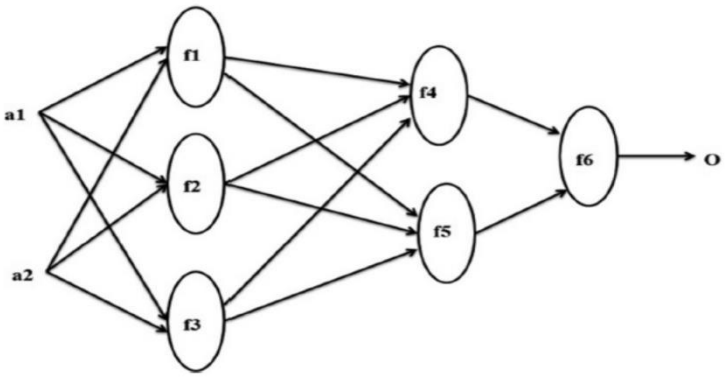


Figure 1.11. Feed-forward method in BPNN [61]

- **Classifier and alert mechanism.** This component is the one that decides whether the information or data received from the activity monitor is malicious or benign.

The proposed method is implemented using the mininet software and the RYU controller since it's customized and designed to embed SDIoT-IDS. The Figure 1.12 below depicts the flood attack mitigation process as an example experiment.

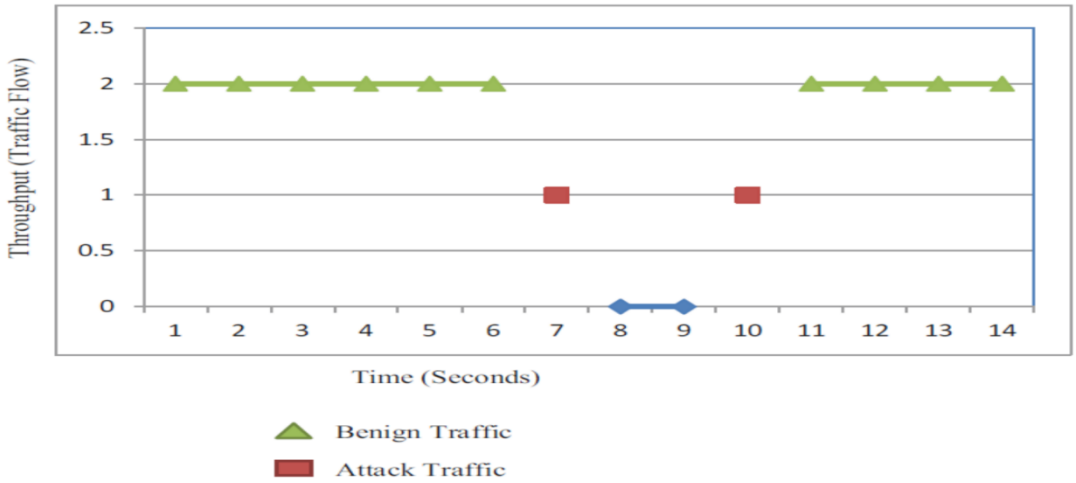


Figure 1.12. Flood attack mitigation by SDIoT-IDS [61]

Positives and Negatives.

Positives: the proposed mechanism proves to be effective against some of the main threats like, distributed-denial of service attacks (DDoS), data theft and botnets, since it is successfully able to detect anomalies and preserve network stability by directing malicious traffic away from the targeted device and network nodes according to flow rules stated in the tables which are generated by the controller, so, it could promote the system performance to be more capable of controlling and blocking various malicious threats and potential attacks.

Negatives: based on this research, the traffic first will be routed into an SDN gateway that will analyze it, but according to the research the activity monitor will be placed at the IoT gateway and that's (an openflow switch), so in other words will it receive the traffic before the router of SDN (gateway)? if so, then that means that the traffic will be routed to it before the router, and that means that it will be vulnerable to any attack with no router to route the traffic to the controller first, and even if not then what happens if the attack was conducted against the openflow switch itself not the router? Especially that it has only traffic analyzer, so if an attack disrupts the switch or takes control over it, then it could manipulate the whole mechanism and allow malicious traffic to pass and maybe then send some traffic to generate whole new flow rules in the flow table.

Related works.

Regarding this topic many researches have been done to improve the field of protecting IoT, like what [62] has done by proposing an IDS for detecting internal anomaly in IoT environment, while [63] suggested an IDS for wormhole attacks infecting the IoT devices.

1.3.5 Developing WSAN structure using SDN

Wireless sensor and actuator networks (WSANs) face challenges in flexibility and configuration. SDN, as a programmable network, offers a solution. This research proposes WSANFlow, an interface protocol enabling communication between the SDN controller and end devices to measure throughput, delay, and power consumption, comparing results with the ZigBee-based model.

WSAN framework and WSANFlow.

This work suggests to use the WSANFlow to be the messaging protocol between the SDNC and the Eds. The suggested controller architecture is built on top of the IEEE 802.15.4 Datalink/MAC layer of the available sensor network protocol. The SDNC performs many duties like gathering statuses where Every SDN-oriented ED sends its own status information to the SDNC. Now regarding the SDN-based end device Ed, it's inferred that according to the research, it can transmit, receive and forward data streams. The SDNC architecture is shown in the Figure 1.13.

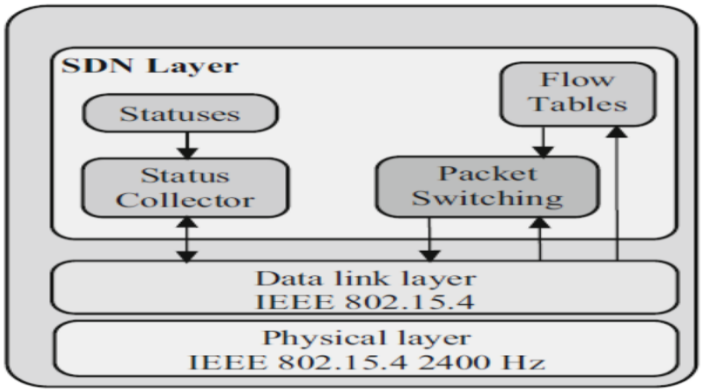


Figure 1.13. WSANFlow protocol architecture [64]

The research proposed a topology for how the framework works and it is shown in the Figure 1.14 to depict that typical scenario.

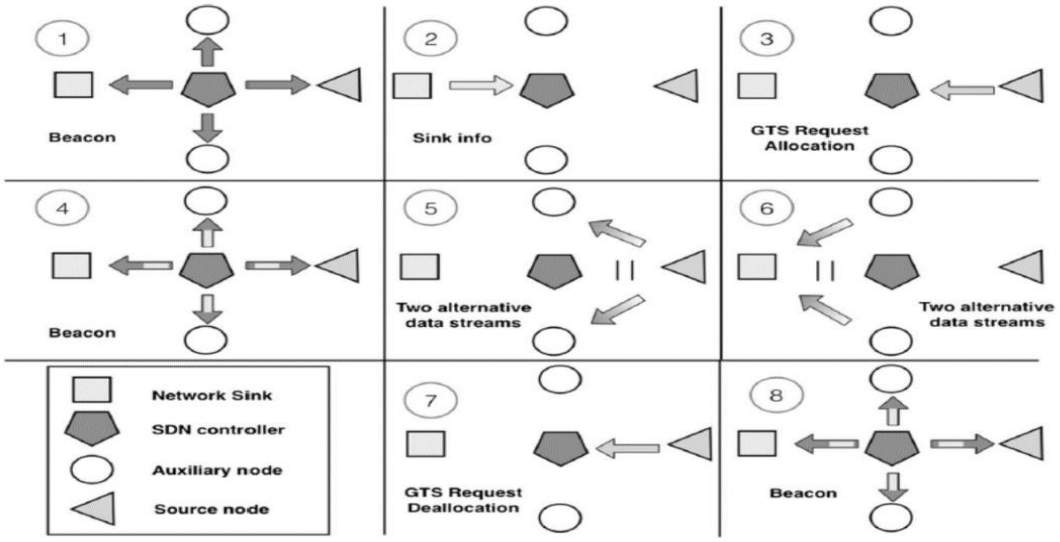


Figure 1.14. Typical scenario for the proposed framework [64]

Positives and negatives.

Positives: WSANFlow introduces a better management method for the WSAN infrastructure through the usage of SDN technology and reduces the power consumption ratio of the network.

Negatives: the researchers did well in their research but still they didn't explain the difference between OpenFlow and the proposed WSANFlow protocol or make a Comparison between them to let us understand why WSANFlow is better, even if it prevails on the ZigBee-based WSAN system, there wasn’t gained any percentage of difference in performance success. Also, the proposed framework contains three main components the controller, end device (ED) and the communication interface (WSANFlow), so what happens if that topology changes and there is horizontal SDN controller topology where there are have multiple controllers working together or as it's called SDN east-west interface? How can the proposed framework deal with that topology, especially that SDN topology is dynamic so, despite that this framework is applied to WSAN environment but it can't be applied to the dynamic topology of SDN as well.

Related Work.

There many works regarding SDN but few regarding the approach of using a WSAN in the context of SDN. But still there are some works in that field like, [65] in their research scientists proposed a three-layer software-defined wireless sensor network architecture with obvious separation of data and control planes. In other research [66], the suggestion was to use SDN in WSN in the context of a generic controller mode, providing emphasizing for general functions like tracking algorithms. While in [67] the MATLAB tool was used to show the

efficiency of openflow-based WSN architecture. A conference paper; like this one [68], presented a decrement method for the data transferred between the controller and end devices.

1.3.6 The design of SDN energy-aware traffic services testbeds

This research discusses building energy-efficient testbeds [69] that could be used for evaluation of performance of energy-aware traffic engineering strategies, also this work discusses the two proof-of-concept testbeds where one of them uses the SDN services based on the open-network OS and the other one uses a system containing open vSwitches and RYU SDN controller. The purpose of these testbeds is to validate previously suggested energy-aware traffic engineering approaches working in different environments.

Software Defined Networks and Energy-aware traffic engineering.

The traffic engineering field helps enhancing the routing function in the network hence, the general performance of the network with optimization of the available system resources. Due to the growing need to get full measurement information and results reading for the state of every element and device in the network, information like needs of applications and source of electricity powering the network elements; the SDN was the new technology that could help developing the traffic engineering. So, the two main testbed designs will be:

- GrEen Traffic engineering testbed: segment routing (GETB-SR). In this design, software integrates the proposed platform with the Grid5000 testbed. Users request node clusters as virtual switches or traffic sources via an application, requiring an OS image and network topology. The Figure 1.15 below shows the segment routing platform's architecture.

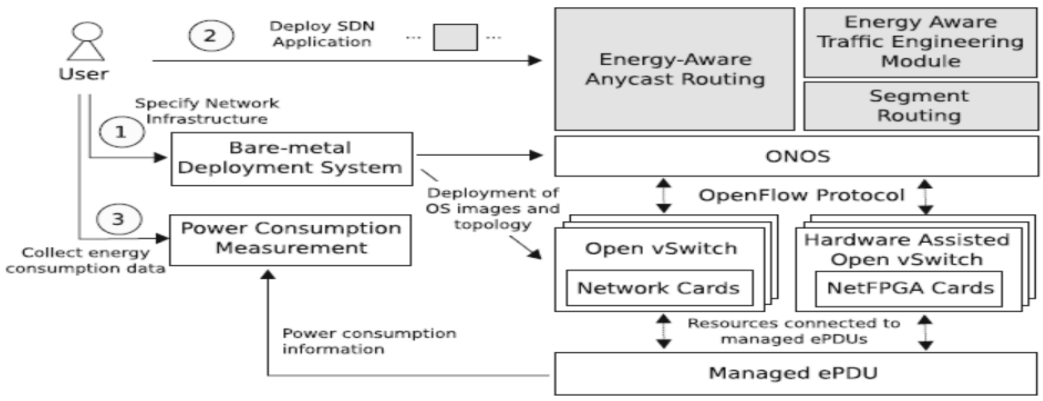


Figure 1.15. (GETB-SR) platform [69]

- GrEen Traffic engineering testBed: anycast routing (GETB-AR). This platform ran on a server with an SDN controller (2048 MB vRAM), 14 vSwitches, and 14 machines simulating data center services. All testbed traffic remained within a single server. A RYU controller handled energy-aware anycast routing using real-time network data, topology, user

input (via REST API), and other control information. The Figure 1.16 below shows the architecture of anycast routing platform.

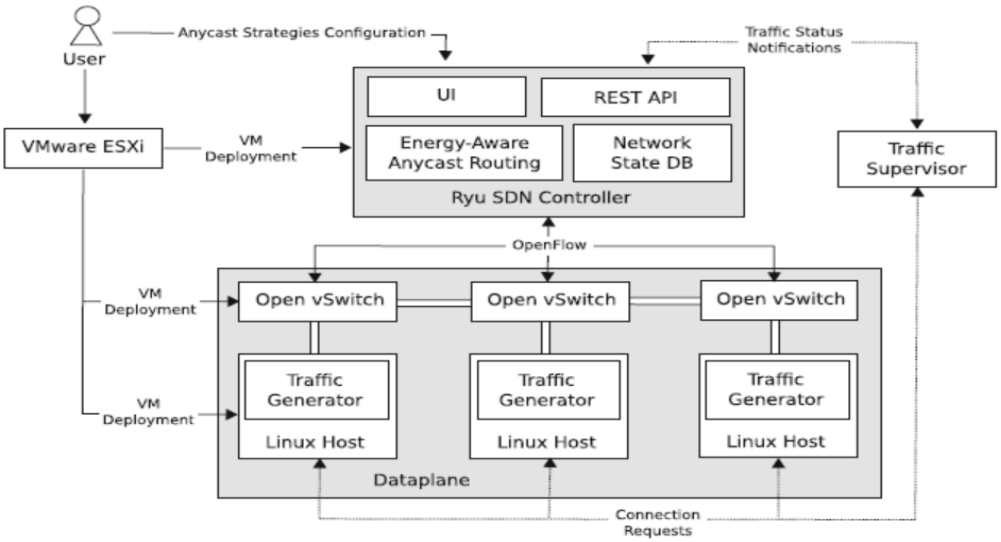


Figure 1.16. GETB-AR platform [69]

Positives and negatives.

Positives: SDN has a better security effect and using it to design a better energy-aware traffic will create a better network routing functionality, energy saving, which will empower the network reliability and the overall performance.

Negatives: the proposed platforms especially the second one (GETB-AR) requires lots of equipment and hardware.

Related work.

Similar researches to this work could be, putting network interfaces into sleep mode as mentioned in [70] or changing flow paths to make increments in idle periods of certain links like what was mentioned in the work of [71]. A green data center testbed has been proposed in the work [72].

1.3.7 The usage of SDN methods to detect and mitigate distributed denial of service (DDoS) attacks

This work [73] classifies DDoS detection techniques and proposes ProDefense, an SDN-based defense mechanism that generates security alerts based on application needs. Designed for large-scale networks, it enables application-specific DDoS detection and mitigation, aiming to categorize solutions and leverage SDN security for ProDefense implementation. The Figure 1.17 below depicts the how ProDefense works.

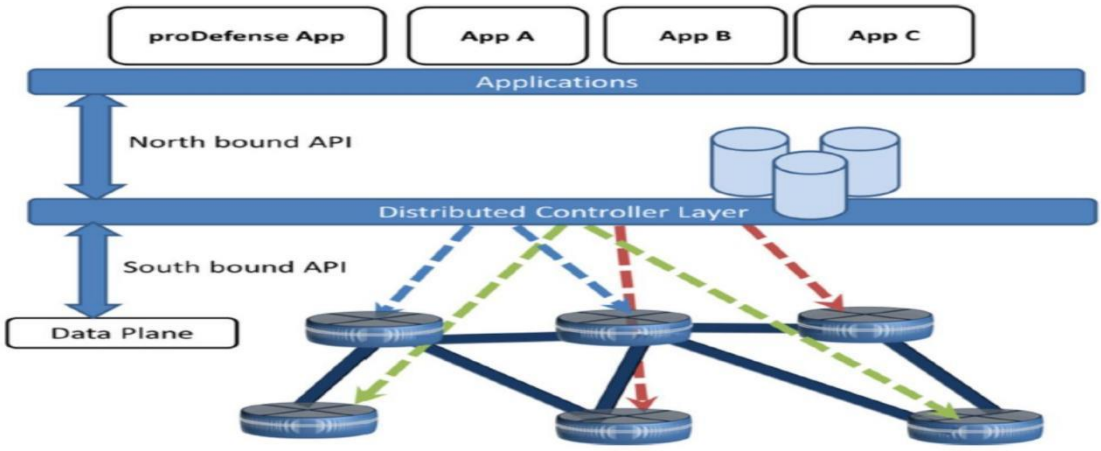


Figure 1.17. ProDefense work design [73]

The researchers here try to divide the applications and services provided by the smart city into three main categories which are: moderate, critical and catastrophic based on their cyber security needs; there are some samples of these applications in Table 1.1.

Table 1.1. Categories of Smart City Applications and Related Security Aspects [73]

Service	DDoS attack effect	Service security requirements	Security solution need	Filters of ProDefense
Traffic control system, smart grid.	Catastrophic	These applications immediately trigger the mitigation system. monitoring the malicious behavior the alarm will be generated before Reaching the threshold.	Such types of apps need very low false-negative rate of threats detection, since the attack's impact is highly critical (extremely agile).	Uses Highly Reactive filter
Location-based and healthcare services	Critical effect	After a specific period of time from the attack detection, the mitigation solution will be triggered	Since the effect of the attack is critical then a delicate solution will be needed.	Uses Intermediate Reactive filer.
Parking and weather update news.	Marginal impact	With this type of apps, it's hard to afford to block legitimate traffic, so first there should be a confirmation for the attack then the mitigation technique will be triggered.	This type of applications requires very low false-positive rate of attack detection, Due to the importance of continuity of such services (agile).	Uses Low Reactive filter

The ProDefense suggested framework uses the exponentially weighted moving average (EWMA) filters to create a customizable solution for attack detection, as in the equation below:

$$PT_t = aPT_{t-1} + (1- a)\, CT_t + c. \quad (1.6)$$

Where PT_t represents the predicted traffic while CT_t refers to the current traffic, α refers to the gain and c is the constant dependent on traffic characteristics. So, it would mean that these filters

will have different reactions from the fastest to the slowest depending on the value of the gain where less gain triggers faster DDoS alerts, so there are three main filters: highly reactive, intermediate reactive and low reactive for the three main categories catastrophic, critical and moderate respectively. The Figure 1.18 below shows the architecture of ProDefense.

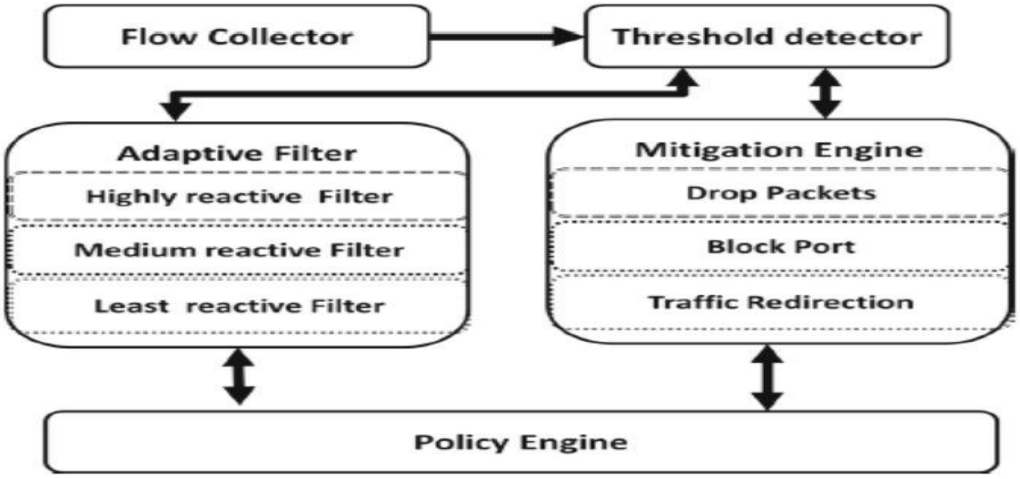


Figure 1.18. ProDefense framework [73]

Positives and negatives:

Positives: Talking about the upsides is so easy since it has many benefits like creating a solution that can satisfy some application-specific requirements of detection and mitigation of DDoS attacks to protect the cyber infrastructure of smart city context.

Negatives: using SDN makes the detection of flow-based traffic flooding attacks easier but, there could be a weakness point in that structure which is attacking the controller itself since it's the central control point over the whole network, such as launching DoS denial-of-service, so there’s a need for controller protection.

Related work.

Some related works to this one are: the work of CCTV [74], it is seen that it discusses the discovery of a botnet that is based on CCTV to create DDoS attacks, the work is about the Bank of Greece website attack [75] where it introduces some information about the attack on the Bank of Greece website which neutralized the servers for 6 hours, and the spamhaus [76] where it documents the 300 Gbps DDoS attack on the organization of spam-fighting.

1.4 General comparison of State-of the Art works and formulation of tasks and algorithms

Now let's have a wrap up comparison for all the previously mentioned methods and algorithms of different works regardless of their relevance to each other but the important thing

is their relevance and effect on SDN environment or their usage of SDN's technology. Table 1.2 shows that.

Table 1.2. General Comparison between SDN-related Methods and Technologies

The method or algorithm	Positive effect	Negative effect
OpenLL	It has better accuracy than OpenNetMon that it's based on	It uses active method for measuring which will affect and may manipulate the network behavior negatively
OBSERVER	It has better results and accuracy in detecting time channel attacks	It needs the resources available in the cloud which means that the SDN has to be permanently connected to the cloud which is not solidly secure environment itself
SDN as a home network	It's a good database for SDN home environment	It lacks the detailed explanation of the main algorithms mentioned in it
SDIoT-IDS	Proves to be effective against anomalies, like cyber theft, DDoS and malware viruses	The activity monitor is mounted on an OpenFlow switch which could be a weakness point
WSANFlow	introduces a better management method for the WSAN infrastructure through the usage of SDN technology, and reduces the power consumption ratio of the network	No recognizing feature from OpenFlow, also it might not suit the dynamic topology of SDN
GETB-SR And GETB-AR	It leverages SDN to design a better energy-aware traffic that will create a better network routing functionality, more efficient usability of resources, energy saving	Requires lots of equipment, which could be an obstacle
ProDefense	It can satisfy an application-specific requirements of detection and mitigation of DDoS attacks to protect the cyber infrastructure of smart city context.	Central control and management point could be under threat which might threaten the whole network environment

This comparison is based on the positives and negatives noted in every work mentioned here, the most important thing to be noted from this works citation and their comparison is the importance of software-defined network and its ability to be integrated in different technology aspects. And last but not least here are some ideas that some could be used in the upcoming chapters, in pursuit to develop the SDN security level using one or some of the following ideas:

1. The Hydra framework, where it can help create an SDN controller software or application that can be downloaded on many or all network smart nodes (computers, routers etc.) so in case of an attack on the controller; the controller gets isolated and another network node like PC, server, or hardware-based firewall could be elected as the controller, based on some

algorithms like priority, next hop, designated router (DR) which is a hardware piece playing a particular role in wireless networking. It is most frequently used as part of an Open Shortest Path first or OSPF link-state routing protocol for IP networks [77]. Hence comes the name Hydra where the network in such a situation that an attack occurred will elect secretly another node to be both the router and the controller without disrupting the work of the network. And that could be applied on both horizontal and vertical controllers' approaches. The application could be installed on all network nodes, whether the network was SDWAN or SDLAN or SDMAN. Also, it could be containing a phone app to monitor other controllers using a main anonymous controller, in case of multiple controllers in use.

2. QoS for Software-defined networking paradigm [78].

3. A protocol that uses hashing algorithm like MD5 [79] or SHA-256 [80] with Blockchain technology [81] to secure the communication between multiple controllers, by designing a java-based framework that can be integrated with openflow to secure east-west communication.

4. Using artificial intelligence to create an analytical algorithm for detecting anomalies based on the statistics, and that algorithm could be built in a framework that is integrated with the controller.

Problems noticed in the SDN environmental structure.

- Centralization: as mentioned before one of the main advantages software-defined networking technology presents is the ability to control the whole network from a single point of management which can be both time and cost-efficient technique but, in the same time from the all the previously mentioned alongside with the combined extrapolation of other researches; it has shown that it could be a single point of failure in the structure of SDN.
- Connection: to solve the above issue some researches tend to use multiple controllers but, in this case, there will be another issue to patch up which is the Application programming interface connection API between those controllers, which is named as the east-westbound API.
- Security level measurement: many researches have tried to measure the level of security of SDN but with the usage of a general network security level equations but, not with a specifically-designed equation to measure the security level of the SDN paradigm and some have used some kind of modeling for their own SDN models and environments.

Important scientific problem to be solved: is elaboration of a new suite of algorithms and SDN controllers' topologies to increase the security level of SDN and elaboration of the theoretical assessment of computer networks' security level.

Objectives to solve the noticed problems in the SDN environmental structure.

1. Analysis of Existing Methods and Technologies for Security Assurance of Computer Networks.

2. Elaboration of Algorithms and Topologies for Assuring the Security of Computer Networks.

3. Efficiency evaluation of the Proposed Topologies for Computer Networks Security Assuring.

1.5 Conclusions of Chapter 1

1. It is provided here an explanation of every single layer in the SDN paradigm and their effect and features, then comes a comparison between traditional computer networks and the software-defined networks. After that, it was presented here a complete analysis of the SDN effect in different aspects, in terms of security, economy etc. It was shown that SDN is the new trend in technology development. Also, SDN is a cost and time effective solution: with SDN it's possible to configure an enterprise network of hundreds of switches with a push of a button according to admin's own requirements

2. It was presented a security efficiency evaluation and an analysis of some of the most prominent SDN-related techniques, researches and methods, was made which shows the importance of SDN in the technology and its ability to be incorporated with other technologies to enhance them and the flexibility of SDN to accept other technologies in its paradigm to gain more enhancements.

3. It was done a general comparison between the different approaches in terms of positives and negatives based on their relation to the software-defined networks technology and their security effect.

4. The given analysis and comparison, permitted to formulate the problems noticed in the SDN environmental structure, important scientific problems in the SDN paradigm and the ways to solve them. Those problems are the centralization issue, the east-westbound API and, and the lack of security level assessment tools for the SDN. The solutions that were presented to patch those issues will assure the security of software-defined networks paradigm to facilitate a more agile transition of the networks' technology from the previous classical paradigm of computer networks to the SDN paradigm hence, provide assure a better security level for computer networks through the usage of the SDN technology.

2 ELABORATION OF ALGORITHMS AND TOPOLOGIES FOR ASSURING THE SECURITY OF COMPUTER NETWORKS

2.1 Description of the situation

Security has been, is and will always be a necessity in all aspects of life, with the emergence of industrial revolution, the human civilization got a huge advancement and the security aspect got a whole new meaning. With the emergence of the internet, every day's life changed, and now it is needed to secure the internet, since it became a life necessity. The main idea of this research is to assure the security of Software-Defined Networking, to make it a safer environment, to make it the best choice for computer networks management, all that to assure the security of computer networks hence, making the internet environment a safer place against cyber-attacks and to make peoples' life better and easier. To secure the SDN, this study has analyzed some researches that have used SDN with other technologies or integrated other technologies with SDN. This analysis has given an idea about some issues that need to be addressed and patched. Then this study proposed some solutions for the found issues in the SDN paradigm. The suggested solutions by this study are a combination of technologies, algorithms, different SDN controllers' topologies and mathematical tools to evaluate the SDN new enhanced security by those newly proposed algorithms and topologies. This chapter provides a comprehensive description of the proposed solution to consolidate the security of software-defined networks and to assure that it is a safer environment for computer networks management and that solution is represented by the framework presented in this work which contains some algorithms, methodologies and topologies this chapter consists of the following phases:

- Phase 1: after enlisting the most of the prominent attack methods and technologies; comes the defense techniques and that's in chapter one, and one of those solutions is the software-defined networking paradigm so, this chapter starts enlisting and explaining the main proposed algorithms and techniques that work as a one integrated framework to develop and assure the security aspect of that paradigm.
- Phase 2: after that comes the topologies that could work with SDN whether with the usage of the proposed algorithms or without. Alongside with a comparison between them on one hand and between them and the usual ordinary one-controller topology on the other hand.

2.2 Algorithms for SDN Security assuring based on Cryptography

It was proposed some ideas based on the cryptography mostly like the cyphering of the transferred information, the cryptography of TLS and the layering methods in the VPN and hash

functions in the blockchain to serve as algorithms and topologies to be the basis of the proposed security suite, but first, it's imperative to show the reasons and purposes from the development of this framework:

- **Centralization.** Now despite that centralization of SDN architecture is one of the main positive features of SDN and an advantage in SDN over the classical architecture in one hand but on the other hand it represents a potential threat itself in the same time by creating a single point of failure SPOF. Meaning that in case of an attack on a single-controller-network; the whole network structure will be jeopardized if the attack succeeds, since the controller is the only main monitoring and controlling entity that administers the behavior of the network. This research will not attempt to remove the centralization feature from SDN; since it's one of its pros over classical networks but, rather try to achieve better performance and assure the SDN security by conducting a decentralization within the centralization framework, meaning that the control layer will still contain the central brain of the network and will be always able to force new policies but in the same time, a distributed-like decision will be created in some cases like what happens in the Parallel and Hybrid topologies, and that will provide the network administration with freedom and agility to some extent while still being controlled centrally. Now it is worth mentioning that to the best of our knowledge up to this day; there are no known or reliable research that could provide a decentralization factor or measurement.
- **East-westbound API.** There is not much concentration on them, and it could be vulnerable to some cyber-attacks [82] like MITM [7], DDoS or DoS types of attacks [6] especially the protocol-based DDoS attacks.
- **Security level measurement.** This research suggests some security and performance parameters to assess the security level of computer networks in general and especially those that use the proposed models as their basis.

2.2.1 Algorithms' suite integrated in Hydra framework

Theoretical Analysis.

There are some SDN topologies that use multiple controllers and here it was also proposed the usage of multiple controllers whether they were of the same brand or from different vendors but, what distinguishes the proposed topologies from the usual existing ones is the way these controllers interact between each other and the secure communication between them using the algorithms that are going to be listed next which will be integrated with this algorithm, and this is the first proposed algorithm of the algorithms suite and it's named as the Hydra algorithm which creates a whole new way of secure interaction between controllers as mentioned. Usually, many network topologies whether they were physical or logical despite the diversity in their

backup plans to deal with different attacks and in spite of the various deterring methodologies, they do not consider some important aspects of securing the networks which are how network nodes are managed, how they behave and how they're communicating and interacting with each other. Our new era networks should be more flexible, resilient and more capable to deal with cyber risks, high loads, huge amounts of data etc.

Our methodology provides better interaction between controllers and makes them able to react as if they were alive or like they have their own brain or an artificial intelligence to some extent, this methodology is the result of the later described technologies combined as a whole suite.

Now as it's known DoS/DDOS attacks are one of the most common attacks due to two main reasons which are: the low cost and simplicity of conducting them which means the ease of waging an attack and their effectiveness in disrupting the target's asset, that's why there is a need to concentrate most of the efforts in this research to consolidate the security of networks from cyber-attacks but DoS/DDoS ones in precise. The abbreviations of DoS and DDoS attacks stand for denial of service and distributed denial of service attacks respectively.

Based on the TCP/IP model, the suggested framework mostly targets protocol attacks since those attacks are mostly work with TCP, UDP packets using the SYN flood attack, those protocols are operating in the TLS layer. Since that most Hydra protocols like Ipsec of the VPN algorithm are working in the TLS layer plus the already existing SDN OpenFlow protocol which is working in TLS as well. Then, its's possible to say that the proposed framework deals mostly with protocol-based DDoS attacks that are basically targeting the TLS layer.

In DoS attack the attacker uses one computer or machine to conduct his attack by creating as much fake requests as he can to flood the network or server and disrupt its ability to provide services hence, becomes unreachable to its requesting legitimate hosts.

While DDoS attacks are initiated by a network of remotely controlled, well structured, and widely dispersed nodes called Zombies [83]. Explaining in details; the case of a DDoS attack the perpetrator tries to make a host or network resource unavailable to its legitimate users by temporarily or indefinitely disrupting services of a machine connected to the Internet like server by flooding it or any other targeted machine or resource with pseudo requests to overload systems and prevent legitimate requests or at least some of them from being fulfilled but, here the only difference is that the incoming traffic flooding the victim originates from many different sources which are simply host machines that could be his or spread around the world in different places and injected with malicious programs called bots or botnets (stands for a combination of robot and network) without the knowledge of their owners and these bots

turn the hosts into zombies which are malicious tools in the hands of the perpetrator to conduct DDoS attacks on any target by the choice of the perpetrator himself and he can also choose when to conduct this attack by setting timers to launch the attack where he forces his bots to flood the targeted network or server with superfluous traffic. That would make it effectively impossible to stop the attack or hinder it just by blocking a single source. A good analogy for DoS/DDoS attack is a group of people crowding at the doorstep of a shop, which makes it hard for legitimate clients to enter, hence disrupting the trade. Criminal DoS/DDoS attackers often aim for sites or services which are hosted on high-profile web servers like credit card payment gateways or banks. Some motivations of these attacks could be revenge, blackmail or activism. There are different tools to create or conduct a DoS/DDoS attacks for example in some cases like Slowloris and MyDoom where the tools are malware-embedded, and launch their attacks secretly with no knowledge of owner of the system. Also, there's this Stacheldraht which is usually used as a classic tool for creating a DDoS attack. This layered structure tool is used by the attacker where he connects the handlers by leveraging a client program, those handlers are compromised systems that issue commands to the zombie agents, that will facilitate the DDoS attack. The first step for an attacker is to build the attack distributed network that could consist of thousands of bot-infected computers (attacking hosts, bots, or zombies) Agents are compromised via the handlers by the attacker, with the usage of automated routines he can exploit some vulnerabilities or weakness points in programs which can accept remote connections running on the targeted remote machines. By the way a handler can control up to a thousand agents.

Sometimes the host machine that will be used as a zombie could be used with the knowledge and consent of its owner like the Operation Payback example, which was organized by the famous group Anonymous, and there are some applications for DDoS attack that were used in this kind of attacks (with the knowledge of the zombie owner) such as LOIC which stands for Low Orbit Ion Cannon, and HOIC was used this way as well, it stands for High Orbit Ion Cannon. Nowadays there is a big selection of DDoS applications that could be used as a threat tool with diverse properties and features, of course some of them are free and some are not and others can be sold in the black market in hacker forums and other related sites.

In summary it is possible to describe the Hydra algorithm's behavior as follow:

1. SDN Topology (any of the proposed topologies).
2. Hydra framework-working normally.
3. Hydra deploys botnets in all nodes connected to the controllers that have Hydra.
4. Connection through VPN tunnel.
5. Encrypting the connection using Double RSA.

6. Block chaining the exchanged data.
7. In case of a DoS/DDoS attack on one of the controllers.
    a. Hydra isolates the Controller IP using ACL.
    b. Hydra activates botnets as a counter measurement against the DoS/DDoS attacker.
    c. Maintaining and restoring the infected controller.
8. Hydra manages SDN topology using the remaining controllers.
9. Reconnecting the restored controller.
10. Getting back to the usual which is, reconnecting the restored controller.

The flowchart diagram of the Hydra is shown in the Figure 2.1.

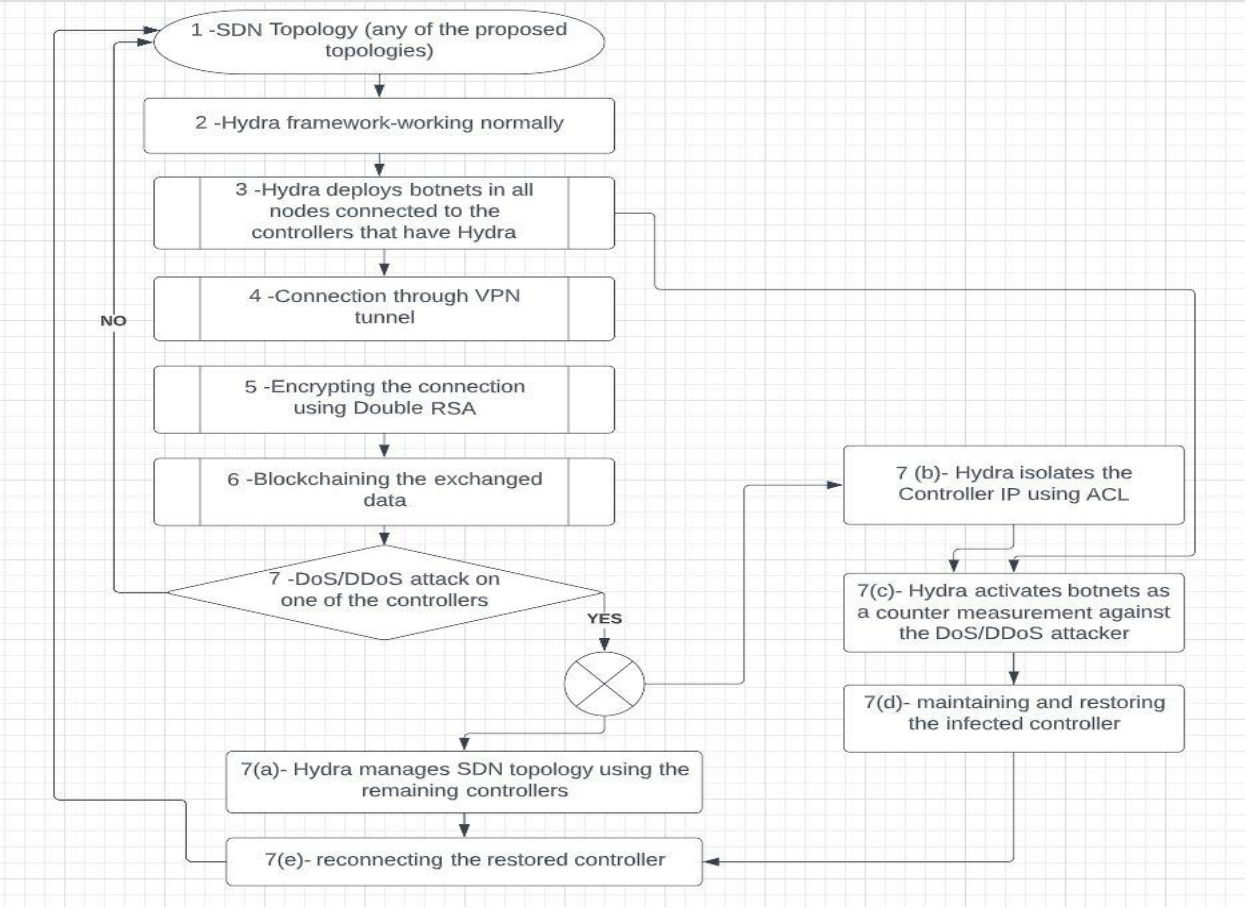


Figure 2.1. Flowchart diagram of Hydra Framework

Practical Analysis.

Again, it is needed to mention that, based on the TCP/IP model, our framework mostly targets mostly DDoS protocol attacks since those attacks are mostly work with TCP, UDP packets using the SYN flood attack, those protocols are operating in the TLS layer. Since that most Hydra protocols like Ipsec of the VPN algorithm are working in the TLS layer plus the already existing SDN OpenFlow protocol which is working in TLS as well. Then, its's possible

to say that the proposed framework deals mostly with protocol-based DDoS attacks that are basically targeting the TLS layer.

Hydra is not an application or a hardware device but it's more like a way of management to run and move things in the controller network topology, it's a whole framework consisting of a suite of applications and specific controller topologies to work with them so it's just the result of all the previously mentioned. The Hydra methodology uses the same concept of the DoS/DDoS attacker against him or them; and that's by adding a botnet program in the controller topology to be added to every computer connecting to any of the controllers in the topology. Now in case of an attack the attacker's IP or Ips will be isolated and conduct a counter attack on the attacker using all the computers connected to our network as zombies to disrupt his zombies and assets to stop them forever or at least for a while using his IP as the target whether that IP was for one device or a network; of course if the attacker was able to stop or infect one of the controllers in the network then the Hydra will isolate that controller as well till that controller gets restored.

The Hydra is capable of working with those proposed topologies; with the serial and hybrid topologies the controllers are divided into main and backup ones and the Hydra replaces the infected ones with the backup ones while, with the parallel topology it can let the remaining controllers take place of the infected one till it is restored. Hence, comes the terminology Hydra like the small fresh-water creature, the one when you cut one of its heads another one grows up due to the creature's ability to regenerate its cells; it can even regenerate itself again if you cut it into pieces.

Defense against attack on serial topology.

Now going back to our framework; in the first topology there are 3 controllers 1 is the main controller based on a priority number the less the number the higher the priority is and 2 backup or secondary controllers. When the attacker commits his attack, if he succeeds to stop the main controller then our first backup controller which comes second in the priority, takes lead directly in case of a DDoS or DoS attack on the main controller (main controller from backups will be differentiated using priority number) in case of serial topology which is the first topology. Now there will be a new elected controller for the whole network which could be anywhere inside the same premises or in another country, which will make it harder now for the attacker to continue his attack due to the following reasons:

- There is another controller that will keep the network running.
- The secondary controller which becomes the new main controller, the disruption of

the previous main controller will be taken as an alert to block the infected controller alongside the IP or IPs of the attacker.

- Since the attacker will be blocked then, he can't continue his attack.
- Also, he doesn't know the IP or location of the new main controller and how many backup controllers are there in the topology which will make continuing the attack useless.

There will be a botnet in the proposed framework regardless of the topology used so that after blocking the IP of the attacker and the infected controller, the attackers source IP will be counter attacked by the SDN while all controllers in the topology will be alerted of an attack and the new controller which was a backup or secondary controller, will be directly elected as the main controlling entity so, the attack of the attacker even if it was able to infect the main controller, it will trigger an alert in the environment so, the attacker will now lose the opportunity to continue the attack not just because his IP will be blocked but also because the environment will have a whole new main controller that was backup previously and it's info will be unknown to the attacker and hence comes the Hydra-like behavior.

Defense against attack on parallel topology.

In this topology there are 3 controllers as well but, here they differ from the previous one in the way they interact with each other since they are all connected and integrated as a whole entity like one distributed controller with 3 nodes. So, they have no priority number and there's no backup controller but the nodes will try to back each other up in case if one node was busy then the switches send requests to it will be served by the other closest node; of course the priority of serving switches will be based on distance between nodes and switches and distance will be based on the number of hops. Now Hydra might not work efficiently here due to the weakness point in this topology which is that all controllers work together as one main controller so nothing is being held back which means that in case of an attack the targeted controller will be disrupted and the others will try to back it up by taking as much requests as they can to increase the capacity of the infected controller to support its ability to defend against the DoS/DDoS attacks but despite that it's possible to block the attacker's IP, this may jeopardize the whole network by exposing all the controllers to the attack and if the attack was strong and fast enough it might disrupt the whole distributed controller topology which means stopping the whole network. That's why it is theoretically supposed that this topology might not be able to implement the Hydra-like behavior.

Defense against attack on hybrid topology.

Here the network topology will be of a mixture of both the previous ones and will consist of six controllers, three main ones which will work together as a whole system of distributed

controller nodes and every one of those controllers will have its own backup controller to replace it in the main level in case of an attack. The controllers will be working normally and each one of them will send a copy of its configurations as update messages to its own backup controller to make the latter ready all the time to replace its main controller. Now in case of an attack; if one of the main controllers couldn't defend against a strong comprehensive attack then, first other controllers will try to offer a larger space to deal with the superfluous requests originating from the attacker's machine or machines and if there was no use of that then, the attacker's IP will be blocked but, in case if there was another type of attack and succeeded to disrupt or infect the controller then the infected controller's IP will be isolated and blocked alongside with the attacker's IP, the backup controller of the infected one will replace the infected one afterwards and the whole network will continue its work as usual till the maintenance of the infected controller finishes but, meanwhile the backup controller will not have a backup one so, in case of an attack the first methodology of defense will be used which is extra spare space offered by other controllers and the same thing happens afterwards if that methodology fails then, there are still four other controllers two main ones and two backups hence, multiple Hydra heads. The Figure 2.2 below shows a depiction of the controller behavior under the management of the Hydra framework.

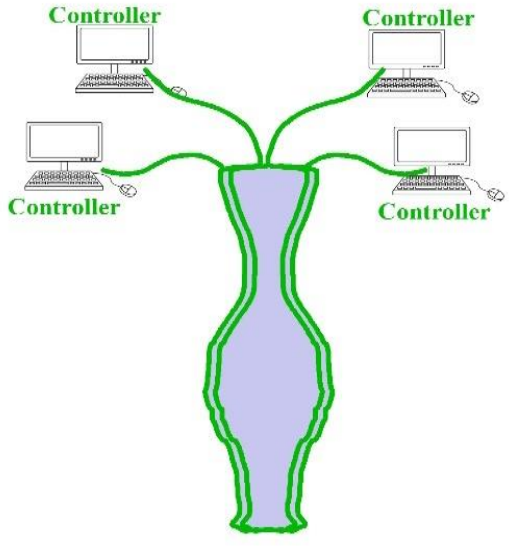


Figure 2.2. The Hydra-like controller behavior

2.2.2 Secured channel of VPN algorithm

Theoretical Analysis.

VPN which stands for virtual private network which is a technology developed to assure the security of a connection channel between two nodes and mainly to assure secure communication between users located remotely from their companies and branch offices and those offices by using a layer tunneling protocol technology alongside with authentication techniques like certificates and passwords to grant access to the virtual private network. VPN has other benefits like securing transactions for internet users since this technology is able to secure personal users' identities and circumvent the censorship and geo-restrictions. A VPN is needed to specify a certainty that the confidentiality of sensitive data can be kept transmitted on the network a Local Area Network (LAN) or workable so that only authorized users are able to access sensitive data [84]. Internet Protocol Security (IPsec) and Secure Socket Layer (SSL) are the two dominant VPN technologies being used today [85]. VPN is able to create a channel-like connection between a private network and another one, which means it provides extension for the private network and that will help users exchange information like if they were all in the same private network or helps them access the intranet of their own corporations while they're home or outside their companies' networks geographically. The application scope of VPN is increasing day by day as the organizations are creating private networks through public Internet using VPN tunneling instead of leased line [86]. Due to ease of management of private networks, functionality and security VPN provides, this gives it many benefits for individuals, corporations and even apps running on end users' devices. Now before going further through the types of VPN, it should be known that the beginning of VPN technology was implemented using different old technologies like dial-up modem, leased line connections that utilized Frame Relay, X.25 and Asynchronous Transfer Mode (ATM) virtual circuits, provided through networks owned and operated by telecommunication carriers. It is worth mentioning that A VPN consists of four main components: 1) a VPN client, 2) a Network Access Server (NAS), 3) a tunnel terminating device or VPN server, 4) a VPN protocol [87].

VPNs could be recognized by a host-to-network or remote access connection by a single computer connecting to a network, or by a site-to-site connection that connects two networks.
The VPN systems can be categorized into the following categories:

- The number of how many simultaneous connections.
- The provided security levels.
- The type of connections topology, like network-to-network or site-to-site.
- The protocol for tunneling that is used to tunnel the traffic.

- The tunnel's termination point location, network-provider edge is an example.
- The OSI layer they present to the connecting network, such as Layer 2 circuits or Layer 3 network connectivity.

VPN protocols for remote access, could be categorized as [88]:

- Point to Point Tunneling Protocol (PPTP).
- Layer Two Tunneling Protocol (L2TP).
- Internet Protocol Security (IPsec).
- Secure Socket Layer (SSL).
- Multi-Protocol Label Switching (MPLS).

The Figure 2.3 shows a virtual depiction of the virtual private network.

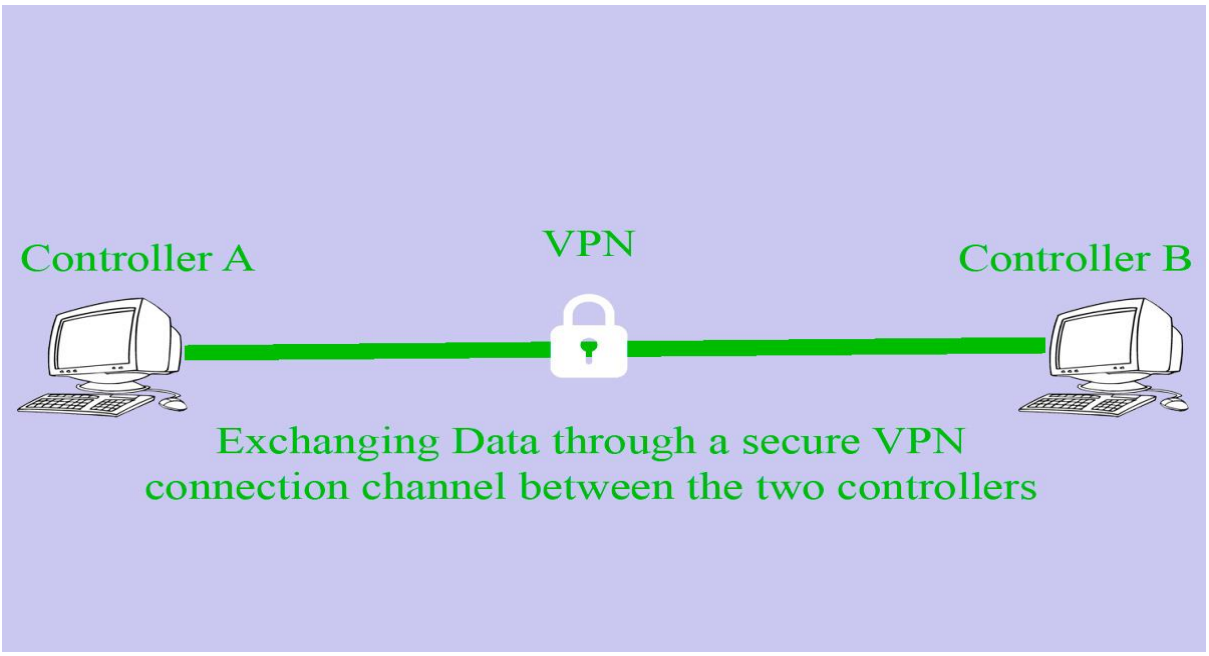


Figure 2.3. Virtual Private Network and its effect on securing the connection

In summary it is possible to describe the VPN algorithm's behavior as follow:

1- VPN working under the management of the Hydra.

2- IPsec Protocol creates a secure tunnel.

3- Encrypting the connection using Double RSA.

4- Block chaining the exchanged data.

5- Attack like DoS/DDoS, MTIM or eavesdropping.

   a. Hydra manages SDN topology using the remaining controllers.
   b. IPsec protocol severs the connection tunnel using the kill switch feature.
   c. Isolation of the attack source if possible.
   d. Maintaining the infected controller and restoring the VPN tunnel.

6- Getting back to the first step.

The flowchart diagram of the VPN is shown in the Figure 2.4.

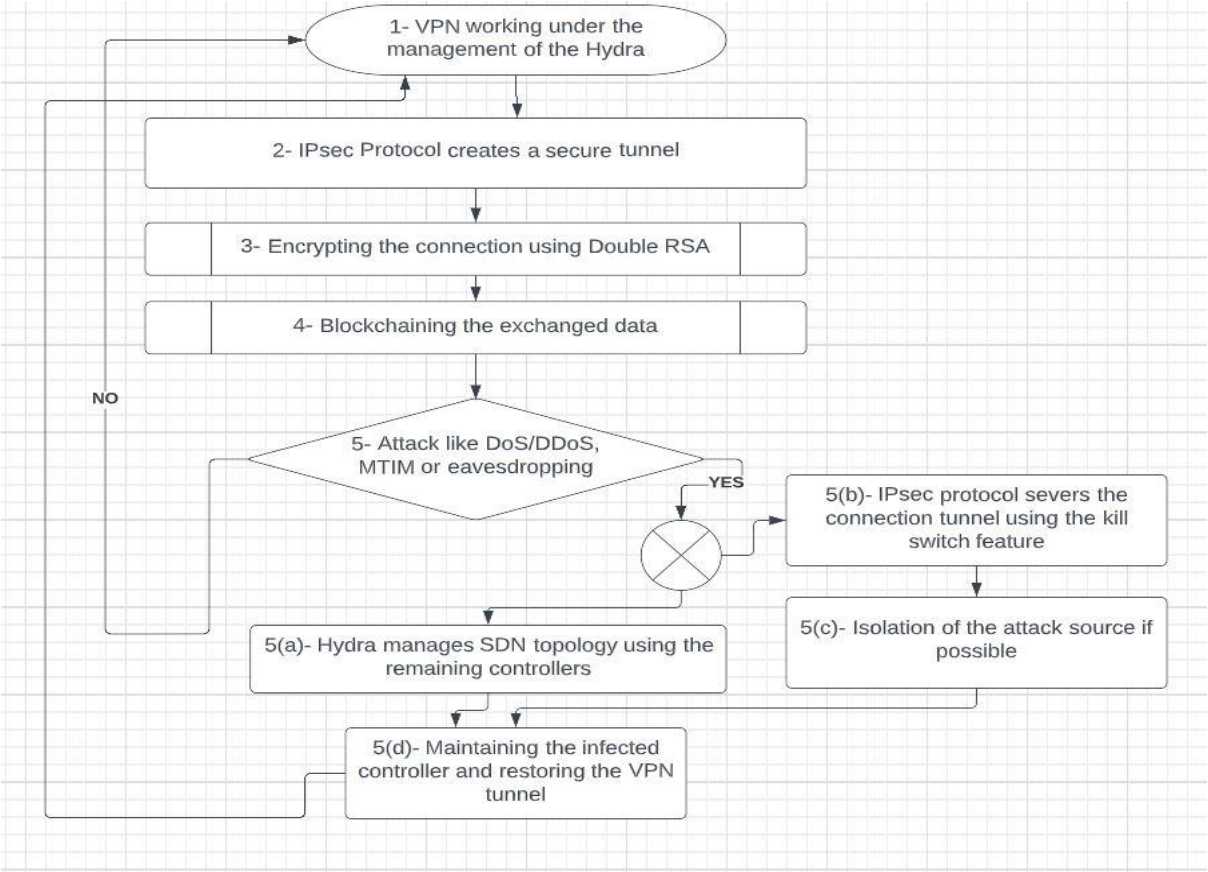


Figure 2.4. Flowchart diagram of the VPN algorithm

Practical Analysis.

VPN established a virtual point to point tunnel-like connection by leveraging the dedicated circuits' technology or by using tunneling protocols through existing networks.

Since Virtual private networking provides tunneling between two endpoints, VPN will be used to create a point to point tunneling between every 2 controllers; because VPN provides a high level of security so, in case of an attack or attempt to break the connection or the virtual tunnel, the connection will be disconnected directly using the kill switch feature and that will alert the whole network of an attack so, it could help against MITM attack. But, in case if the attacker was able to infiltrate the VPN tunneling then, the next algorithm can be used which is RSA. Using this technology might be useful in the future for home Software-defined networks but, not currently since it may require fast internet connection; meanwhile most SDN researches are about data centers and enterprise networks and despite that the usage of VPN may cost time, bandwidth but it won't be a problem to use it in the proposed topologies for the sake of research and because those are proposed for data center to enterprise level environments; especially that many data

centers now have fiber optic connection to the WAN. The Figure 2.5 below shows how the VPN will be used with the SDN environment.

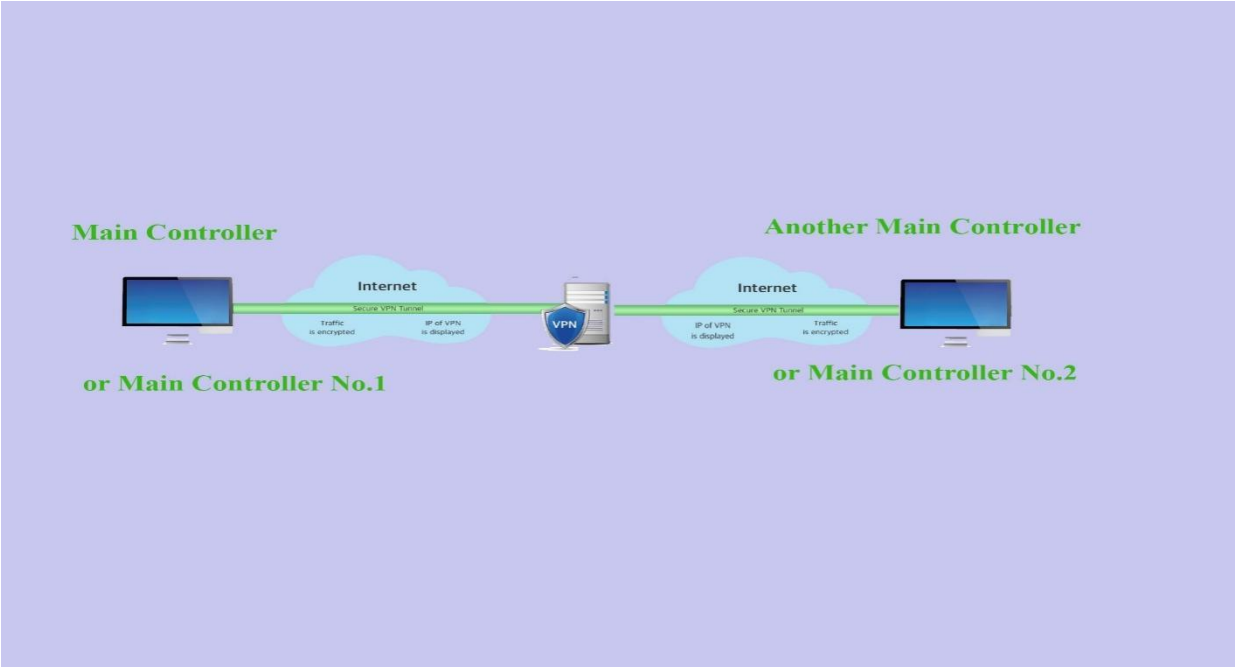


Figure 2.5. VPN usage with SDN controllers

Defense against attack on serial topology.

As mentioned before our proposed SDN controller topologies will be 3, the first one is named the serial topology or controller series topology. There are three controllers, one will be the main controller and the other two are backup controllers, all what they do in the initial state is to receive updates of network configurations and statuses; of course, the meaning of initial state is the state of order in the environment when there's no attack and everything works just fine. As shown in the figure above, there will be a VPN tunnel between every 2 controllers which means that the main controller will connect to a VPN server the internet and through that VPN server the controller will connect to the backup controller No 1 which is the highest in the priority after the main controller and the same thing will happen between the main controller and the backup controller No 2. Now in case of an attack on the controller-controller connection, usually based on the nature of VPN connection will be terminated and that itself will work as an alert mechanism to alert the network of an attack but, in case if that didn't work the framework has an extra two algorithms to work inside the VPN channel for exchange of information between the two controllers so that if the perpetrator succeeded in penetrating the VPN channel which is probably impossible then he will face two other major problems of decrypting the information transferred which is encrypted using RSA algorithm and breaking the blockchain connection made by using the blockchain algorithm.

Defense against attack on parallel topology.

As previously mentioned, here in this environment there are 3 controllers working together as one entity so, we'll have main controller connecting to another main controller through a VPN tunnel connection and the same principles will apply in case of an attack.

Defense against attack on hybrid topology.

Here, we'll have a six controllers' topology containing three main controllers working as one controller with 3 backup controllers for each one of the main controllers. The VPN connection here could be used for both connections between each main controller and its backup one and between each main controller and the other one as well; so, the connection will be here main-main and main-backup.

2.2.3 Double RSA algorithm

Theoretical Analysis.

RSA (Rivest–Shamir–Adleman) is one of the first public-key cryptosystems and is widely used for secure data transmission. In such a cryptosystem, the encryption key is public and it is different from the decryption key which is kept secret (private). In RSA, this asymmetry is based on the practical difficulty of the factorization of the product of two large prime numbers, the "factoring problem". The acronym RSA is made of the initial letters of the surnames of Ron Rivest, Adi Shamir, and Leonard Adleman, who first publicly described the algorithm in 1977. If it is proved to be true that any method to break RSA maybe educe an effective algorithm to factor big integer, it's possible to draw a conclusion that breaking RSA and integer-factor-problem are with the same degree difficulty [89].

In the RSA algorithm the user will generate and publish the public key which is based and consists of two large prime numbers with an auxiliary value. Of course, the prime numbers will be secret. While anyone can use the public key to encrypt a message, the private key will be kept secret of course and that's why it's called private. Preferably the public key will consist of 2 large prime numbers and the one with knowledge of both the prime numbers will be able to decode the message. RSA is not much used to encrypt data because it's slow relatively but, it could be used to pass symmetric cryptography's encrypted shared keys, which will be used in turn as an encoding-decoding operation at higher speeds. This algorithm is also called asymmetric key cryptography algorithm since it uses two different (asymmetric) keys, one for encryption and one for decryption unlike symmetric cryptography systems which use the same key for both encryption and decryption operations.

This algorithm or methodology consists of 4 main steps which are: key generation, key distribution, encryption and decryption. RSA has two categories of keys and they're; the public key and private key. The steps for key generation are given as [90]:

1. Generate two large prime numbers p and q.
2. Compute $n = p \times q$. (2.1)
3. Compute $0z = (p - 1) \times (q - 1)$. (2.2)
4. Let's consider that $mod\ z = v$. (2.3)
5. Choose a number relatively prime to z and call it d.
6. Find e such that $e * d = 1\ mod\ z \rightarrow e*d = 1*v$. (2.4)
7. Public key is (n, e).
8. Private Key is (n, d).

The figure 2.6 shows the RSA algorithm and how it works.

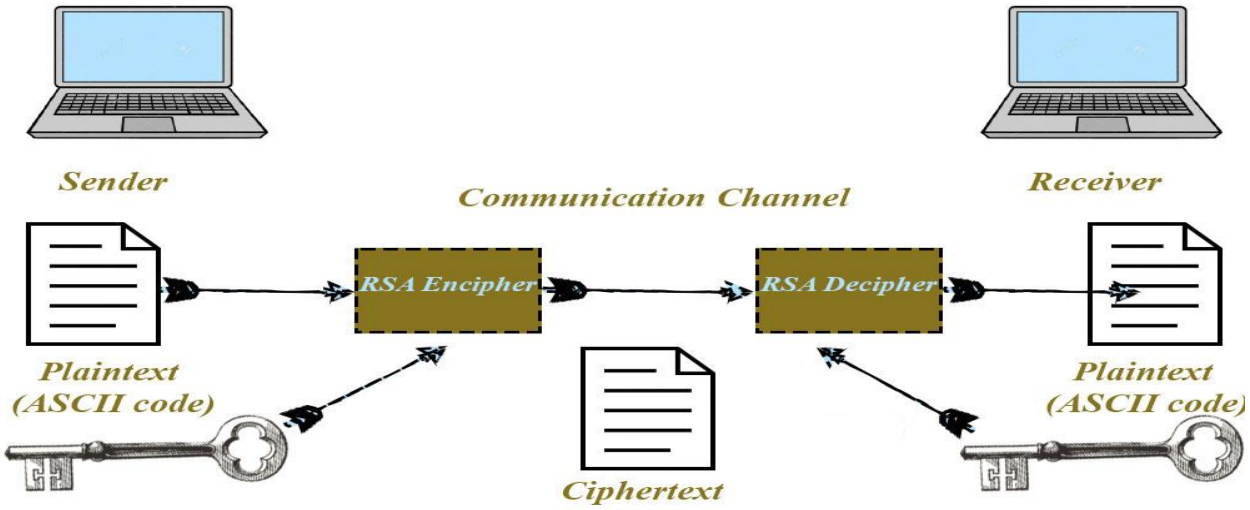


Figure 2.6. Rivest–Shamir–Adleman security algorithm architecture

In summary it is possible to describe the Double RSA algorithm's behavior as follow: Generate Multiple Key Pairs: Generate different RSA key pairs for various purposes, like one for encryption and another for digital signatures. Generating multiple RSA key pairs for different purposes (e.g., one key pair for encryption and another for digital signatures), this approach will not make the RSA algorithm itself slower. Here's why:

1. Algorithm Efficiency: the RSA algorithm's efficiency depends on the size of the keys and the operations (encryption, decryption, signing, verifying), not on the number of key pairs generated. Each key pair operates independently, so generating and using multiple key pairs does not slow down individual RSA operations.
2. Separate Operations: since each key pair is used for different purposes, the operations

are separate. encryption and decryption would use one key pair and signing and verifying would use another key pair.

3. Performance Impact: generating multiple key pairs will take more time initially compared to generating just one. However, this is a one-time setup cost. Regarding the operational speed, during actual use (encryption, decryption, signing, verifying), each operation is performed using one specific key pair, so there's no additional overhead introduced by having multiple key pairs.

4. Resource Management: the main impact of using multiple key pairs would be on resource management, such as, memory and storage where, slightly more memory and storage are needed to store additional keys, but that's not a big problem since they will be mostly used in servers that will work as SDN controllers. One more thing, system Complexity, which means the complexity of managing multiple key pairs could add a small amount of overhead, but this is usually negligible. So, it's possible to understand that using multiple RSA key pairs for different purposes (e.g., encryption and digital signatures) will not make the RSA algorithm slower during actual operations. The only noticeable impact might be during the initial key generation phase, and the need to manage multiple keys. The performance of individual RSA operations remains unaffected.

Digital signatures in RSA play a crucial role in securing communication by ensuring authenticity, integrity, and non-repudiation of messages. Here's how the process works.

1. Signature Creation.

- Message Hashing: The sender first applies a cryptographic hash function (e.g., SHA-256) to the message. This hash function converts the message into a fixed-size string of numbers, which represents the message's content in a condensed form. Even a small change in the message will produce a different hash. Example: If the message is "Hello," the hash might be something like 3a9c....
- Encryption with Private Key (Signing): The sender then encrypts this hash value using their RSA private key. This encrypted hash is called the digital signature. Only the sender's private key can produce this specific signature. Example: The encrypted hash (digital signature) might look like a large number, say 3890482....
- Transmission: The sender sends both the original message and the digital signature to the recipient.

2. Signature Verification.

- Decryption with Public Key: Upon receiving the message and the signature, the

recipient decrypts the digital signature using the sender's public key. This decryption reveals the original hash that the sender generated. Example: The decrypted signature should give the hash 3a9c....

- Hashing the Received Message: The recipient then independently applies the same hash function to the received message. This generates a new hash based on the received message.

Example: The recipient hashes the message "Hello" and gets 3a9c....

- Comparison: The recipient compares the decrypted hash (from the signature) with the newly generated hash (from the received message). If both hashes match, it confirms that:

a. Integrity: The message has not been altered since the sender signed it. Any change in the message would result in a different hash.

b. Authenticity: The message genuinely came from the sender because only the sender's private key could have created the valid signature.

- Non-Repudiation: Since the sender's private key was used to sign the message, the sender cannot later deny having sent the message (non-repudiation).

Why This Secures Communication:

- Authenticity: The recipient can be sure that the message came from the person who holds the private key (the sender), ensuring the identity of the sender.
- Integrity: The recipient can verify that the message was not tampered with during transmission, as any alteration would result in a different hash value.
- Non-Repudiation: The sender cannot deny having sent the message because the digital signature can only be generated by someone who has access to the private key.
- This process ensures secure communication by verifying that the message is authentic and has not been altered, protecting against forgery and tampering.

Clarification.

- Hashing: The hash function itself is not reversible. Once you hash a message, you cannot retrieve the original message from the hash.
- Digital Signature: The digital signature is an encrypted version of the hash (using the sender's private key). The recipient decrypts this signature using the sender's public key to recover the hash.

So, the recipient is not trying to reverse the hash. Instead, they are using the sender's public key to decrypt the digital signature, which gives them the hash. Then, they compare it to their own freshly computed hash of the received message to verify its authenticity and integrity.

The flowchart diagram of the Double RSA is shown in Figure 2.7.

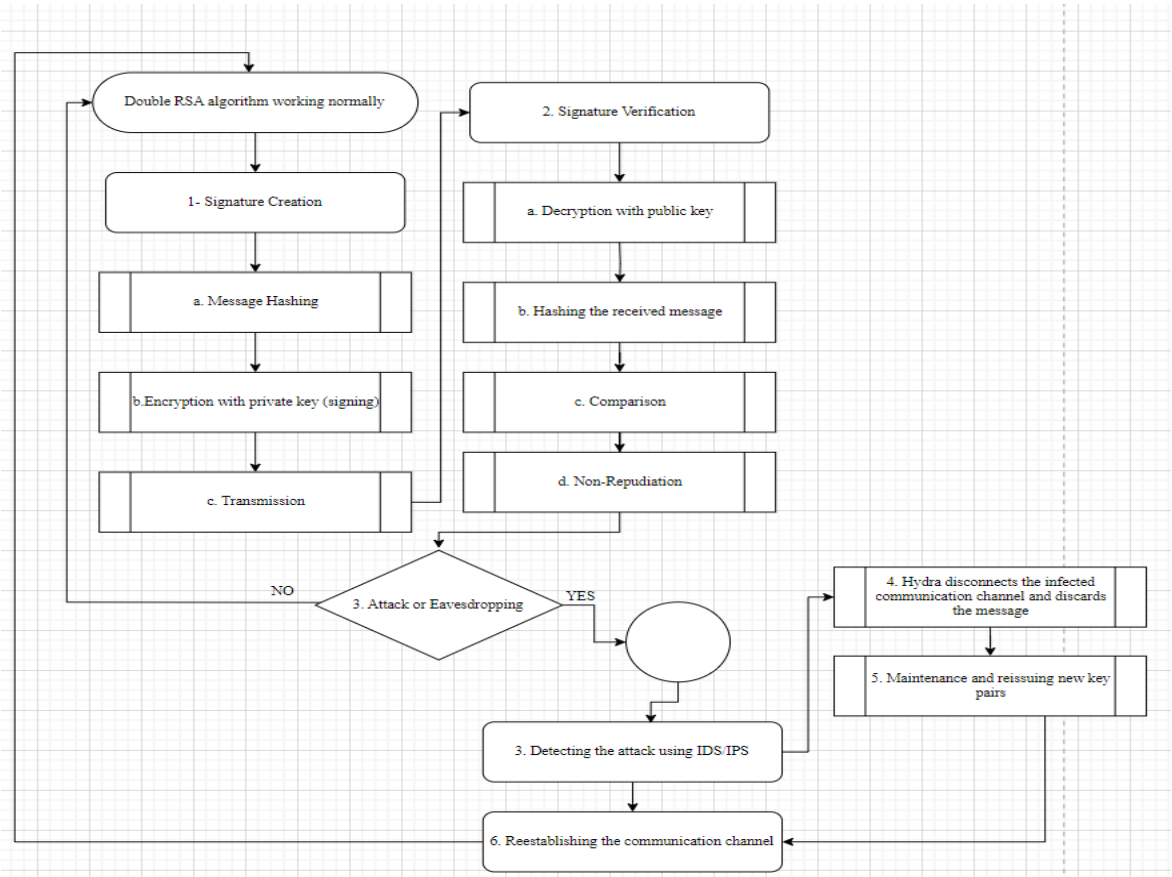


Figure 2.7. Flowchart diagram of the Double RSA algorithm

Practical Analysis.

Inside the tunneling connection the public key RSA algorithm will be used to exchange the keys of connection to start a connection session or for information exchange and the digital signatures will be used for Authentication, Integrity and Non-Repudiation of information. Of course, that's beside the usage of IPsec algorithm used in VPN technology. The usage of double RSA algorithm might a small overload but, it can be explained in the next simplifying diagram. The Figure 2.8 shows the architecture of the proposed Double RSA algorithm.

Figure 2.8. Double RSA algorithms used inside the VPN channel

The Double RSA encrypted connection will be starting by nodes issuing their own key pairs for RSA encryption and another key pairs for digital signatures. Node X will issue the Public Key 1 PU1-X and Private Key 1 PR1-X, the same thing for node Y; PU1-Y and PR1-Y, the group number 1 is for the RSA encryption.

Node X will issue the Public Key 2 PU2-X and Private Key 2 PR2-X, the same thing for node Y; PU2-Y and PR2-Y, the group number 2 is for the digital signatures.

Defense against attack on serial topology. Generally, the RSA will behave the same in all topologies and will be affected in the same way in case of an attack since it works to secure connection between every two nodes so, it doesn't matter if there were three controllers or six controllers or if they were connected in a parallel way or serial one.

Defense against attack on parallel topology. The attacker anyway can't break the algorithm and even if he could that will be after breaking into the VPN tunnel and breaching all those obstacles will take longtime enough to alert the topology of a breach and stopping the attack which will give the network a shorter response time but, just in case if the attacker hypothetically succeeded in conducting a breach; he'll still have to face the next algorithm. Not to forget the digital signatures that will be used for Authentication, Integrity and Non-Repudiation of information.

Defense against attack on hybrid topology. As mentioned before regardless of the controllers' topology, the double RSA algorithm will behave the same in case of normal state for all topologies and in case of an attack it will behave the same with all the three topologies

because it concentrates on the connection between every 2 nodes. Of course, the RSA algorithm will not change anything in its behavior if there was an eavesdropping or listening because it's merely a channel securing mechanism so, it will not alert the admin if there was any kind of listening but, using the digital signatures for Authentication, Integrity and Non-Repudiation of information, alongside monitoring the behavior of the network using IDS/IPS systems will help in attack detection. So, whether the perpetrator was listening to the communication channel between the main controller and its backup or between a main controller and another one it will be the same.

2.2.4 Distributed ledger of Blockchain algorithm

Theoretical Analysis.

The blockchain or block chain, is defined as a growing records' list; that can be named as blocks which are tied digitally using Cryptography. Every block carries a timestamp, a cryptographic hash of the previous block and transaction data. Due to its reliable design, blockchain is considered resistant to altering of the data. It could be described as an open, distributed ledger that has the ability to record transactions between any two parties effectively and reliably. Blockchain is generally administered using a peer-to-peer network that abides by a protocol for inter-node communication and for verifying new blocks where every computer is considered as a network node. Once the data is recorded in a block then it cannot be changed without tampering with all the subsequent blocks, and that would require the network majority consensus. Despite that blockchain records are not unalterable, still blockchains could be considered as a securely designed structure and embody a system of distributed computing with fault tolerance of high Byzantine. A node can be called a miner when it proposes and validates transactions and perform mining to provide consensus to secure the blockchain [91]. So, it is possible to say that blockchain excels in decentralized consensus.

It is believed that Blockchain was developed or invented by a person or more than one person under the name Satoshi Nakamoto in 2008 to be used as a public transaction ledger for the bitcoin which is a type of many types of a new emerging technology called cryptocurrency and that is a virtual currency in the virtual world of internet, which could be the future way of conducting monetary transactions and deals. But Satoshi Nakamoto is unknown. The creation of the blockchain for bitcoin helped to make it as the first digital currency that can solve the double-spending problem with no need for a central server, trusted authority or party. The bitcoin design has inspired other fields of technology, applications, and blockchains which are readable by the public and widely implemented by cryptocurrencies. Blockchain is considered as a kind or category of payment rail. For business usage, private blockchains have been suggested.

Blockchain is a promising technology and works as mentioned before as a public ledger which works like a log by keeping a record of all transactions in a chronological order, secured by an appropriate consensus mechanism and providing an immutable record [92] so, it's possible to say that the blockchain is a decentralized ledger of all transactions across a peer-to-peer network. By leveraging this technology, nodes would be able to verify transactions with no need for any controlling authority to verify and certify. This technology may have many potential applications like settling trades, fund transfers and voting. The bottom line is that Blockchain technology is regarded by many IT innovators and experts as one of the most significant technological innovations in recent years in the field of digitization of secure ownership of assets [93].

Usage fields.

The Blockchain technology could be included in various areas and could be integrated with other miscellaneous technology fields. Currently, blockchains are principally leveraged for cryptocurrencies as a distributed ledger.

Currently Blockchain is being used in some fields like:

- **Smart contracts.** a smart contract is a set of promises, specified in digital form, with a program enforcing the contract built into the code [94].

- **Supply chain.** Many industrial organizations exist and efforts given to employ the technology of blockchains in the supply chain logistics and its management. Supply chain could be defined as a system of people or activities or organizations or resources related to the movement of a service from the provider to consumer or customer. Supply chain management is a critical aspect of conducting any business [95].

- **Some other usage fields.** There are diverse approaches and fields that could use Blockchain technology as their performance's basis for instance, it could be used for issuing a public, permanent, transparent ledger system for tracking digital use and payments sent to content creators, compiling data on the sales. Nowadays there are new distribution methods that exist for insurance industry like parametric insurance, peer-to-peer insurance and micro insurance after the adoption of the technology of blockchain. As mentioned before blockchains could be integrated and incorporated with many other fields because it's a promising technology that's why its effect is noticed on the IoT technology and the sharing economy because they involve many collaborating peers. Another application of blockchain is the online voting.

  - Other designs: there is a service that works online and verifies and validates the existence of computer files as of a specific time and that service is called the Proof of Existence.
  - The decentralized voting called Tezos.

- Quorum which is a private, permissionable blockchain technology by JPMorgan Chase, used for contract applications and it has a private storage.
- Hyperledger is a collaborative effort and a cross-industry from the Linux Foundation to support the distributed ledgers that are based on the blockchain technology, and there are plenty of projects under this initiative like Hyperledger Fabric (spearheaded by IBM) and Hyperledger Burrow (by Monax). There are 4 known categories of blockchain networks and they're: consortium blockchains, private blockchains, public blockchains, and hybrid blockchains:

- **Consortium Blockchain.**

Some of the differences between consortium and public blockchains is that public blockchain has no access restriction, hence, anyone can be a participant in it. Meaning that, anyone in the world has the ability to read the data contained in the blockchain, and not just to read the data but also to issue transactions on a public blockchain. On the other hand, consortium blockchains differ from public one in the permission characteristic because they are considered as permissioned, meaning that, not anyone has the access to it. That's why it's possible to consider consortium blockchains as semi-decentralized. Consortium blockchains possess the security features that are inherent in public blockchains, whilst also allowing for a greater degree of control over the network [96].

- **Public blockchains.**

As mentioned before, public blockchain has no restrictions on access which means that anyone with an access to the internet may participate in the transaction process or take his place as a validator (that means he becomes a part of the consensus protocol execution). One of the biggest public blockchains examples is the public bitcoin blockchain. This type of networks offers some economic incentives for those who provide a better security for it and use any algorithm of a proof of work or proof of stake. Proof-of-Work or PoW, is the original consensus algorithm in a Blockchain networks, where user sends a digital token to each other, verifies the transactions and create new blocks to the chain [97]. It is imperative not to forget to mention that proof of work (PoW) is a measure to deter denial of service attacks and other service abuses such as spam on a network and that's done by requesting some work from the requester of the service, which would mean processing time by a computer. The mechanism of proof of work consensus is the most commonly used mechanism for consensus in the existing blockchains. The PoW technology was proposed by Bitcoin and suggests that every peer will vote with his "computing power" and that's by solving proof of work instances then issuing the suitable blocks. Bitcoin, for example, employs a hash-based PoW which entails finding a nonce value, such that when

hashed with additional block parameters (e.g., a Merkle hash, the previous block hash), the value of the hash has to be smaller than the current target value [98]. But consensus based on PoW is vulnerable to 51%" attacks, a 51% attack may occur when a single miner node, which happens to have exceptionally more computational resources than the rest of the network nodes, dominates the verification and approval of transactions and controls the content of a blockchain [99]. So, in other words proof of work is a type or one way to implement consensus algorithm; the process of having nodes accept a new version of the ledger is commonly referred to as consensus [100] and proof of stake is a type of consensus algorithm by which a cryptocurrency blockchain network aims to achieve distributed consensus. In PoS-based cryptocurrencies the creator of the next block is chosen via various combinations of random selection and wealth or age (i.e., the stake). PoS enables to achieve the consensus via proving the stake ownership [101].

- **Private Blockchains.**

As inferred from the name, private blockchains have access permission. thus, if someone wants to join them, then he has to gain an invitation from the network administrators. Meaning that the access of both the participant and validator is restricted.

- **Hybrid Blockchains.**

As inferred from the name, hybrid blockchain can be simply defined as a combination of both public and private blockchains' merits. One of the prominent characteristics of this type is its permission for users of the blockchain APIs to choose what information can remain private and what information should become public. The figure 2.9 shown; depicts a virtual look of the blockchain methodology.

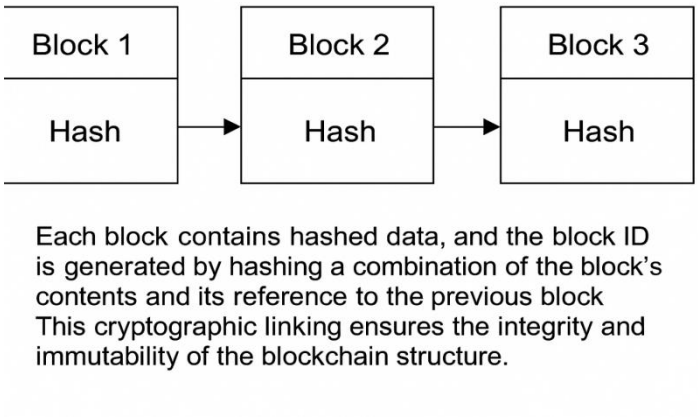


Figure 2.9. Blockchain Technology

In summary it is possible to describe the Double RSA algorithm's behavior as follow:

1. Blockchain working normally regardless of the topology to protect the information integrity.
2. Hashing the networking information.
3. Gathering the hashes of the information with the blocks number and hashing them all.
4. Hashes verification.
5. Issuing the first block.
6. Sending blocks alongside the sent information every 10 seconds.
7. Receiving node will compare the received hashes with the send information after hashing them again comparing the two hashes to check the integrity of the information.
8. Attack or data is tampered with.
    a. Hydra disconnects the Blockchain-based communication that contains unmatching block.
    b. The block that contains unmatched hashes will be dropped.
    c. Maintenance and checking all the coming blocks and performing a cleanup.
9. Reestablishing the communication channel.
10. Going back to the first step.

The flowchart diagram of the distributed ledger of Blockchain is shown in Figure 2.10.

Practical Analysis.

To the best of our knowledge Blockchain's usage is limited in network communications yet and its usage could be of a great deal of advantage to the cyber security field. Blockchain can track information and store data in a chronological fashion and if there's a change in the data stored, its block will not be tampered with or changed but a new block will be created instead, containing information about the adjusted or changed data for example X changed to Y.

Before a block gets added to the chain of blocks; a few things must happen:

- A cryptographic puzzle must be solved and that will create the block.
- Proof-of-work, which is the process of sharing the solution of the puzzle by the computer that solved it with all the nodes of the network.

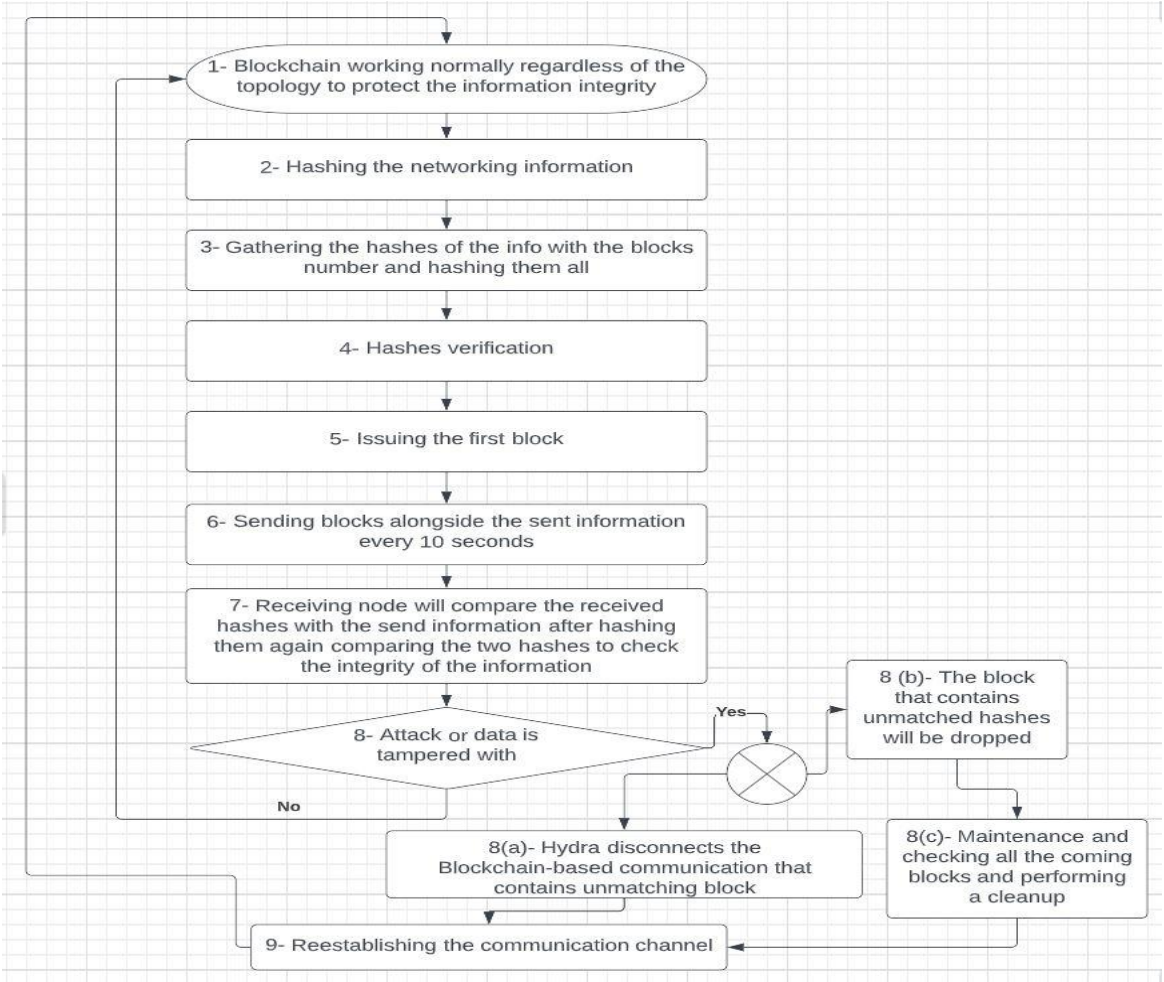


Figure 2.10. Flowchart diagram of the distributed ledger of Blockchain algorithm

- Then the network verifies this proof-of-work and if it was correct then, the newly created block will be added to the chain.

Our framework will leverage blockchain by mining a special blockchain for the network environment and creating blocks of hashes. A hash is just a hexadecimal number created with a fixed length using a hash function like SHA 256 and others. To verify if a block's hash has the right value and the block can be added to the chain or not, a limit will be set or a target number that will be like the limit that the hash value should be less than and not to exceed it. Then a value that represents the difficulty will be created. It is worth mentioning that the difficulty value will be subtracted from the target number which means that the bigger the difficulty number is the less target number we'll have; which means it will be harder for the computer/controller to calculate the block hash value since it has to be less than the target number and that means that the controller will take longer time to guess the right answer hence, it will create more hashes for a block because number of hashes represents the number of attempts done by the device to figure out the right block hash, and that is the meaning of the mining difficulty. Our proposed

framework uses blockchain technology to create a virtual secure channel between the connected nodes using blockchain to hash the information transferred between them; meaning sending that information along with its hashes that information will be containing updates of network configuration and it will be sent every 10 seconds. So, Blockchain could be leveraged for connection authentication.

Since blockchain is an innovative type of database so, there's a chance to use it for both recording and encryption. Every 10 seconds there will be an update of configuration unicasted or broadcasted depending on the topology; from the main controller to the backup controller/controllers, the data sent from the main controller will be hashed and the data will be sent alongside with its hash in an encrypted form; so that the backup controller will be prepared to compare the hashes coming from the main controller next time and to hash the encrypted data again after decrypting them and compare them with already received hash to check if both received and newly issued hashes match which will determine whether that received data are intact or not, and if they were the same data sent originally. In case if they don't match then that means the data were tampered with. Also, the backup controller or the receiving node will check for three things:

1. If the next hashes fit the type of hashes, it has or not, meaning the length of the hash, the difficulty number etc.
2. since any slight change in the data means new hash, if the main controller was attacked and being controlled by a hacker and tried to change the data or update a data of a specific node again and it was with a different length or type of hashes, then it will be rejected by the backup controller after being compared with the previous blocks' hashes and the backup controller will be alerted of a breach in the main controller.
3. Configuration updates have a timeline, for instance at hour 15:00 the ACL or access control list was used to block network 10.10.12.0, at hour 16:00 another network was added to the list let's say 11.11.14.0; by using blockchain which is a distributed ledger that keeps the time order of information and data stored in it and prevents anyone from tampering with it, it will be possible to protect the data from being tampered with by the perpetrator who might try to change the time sequence of those configuration updates; even if he tried to insert a code to manipulate a block let's say block of index 2 and that block contains information about blocking a specific network and changed it into granting access to that network, his new block will not match the chain because it will have different hash from the original one and it will have to have the hash of the previous block and of course, the previous hash contained in this new block is different from the original one since it's a new block that has been tampered with, also not to forget that in

case of any attempt to tamper with any block, that could create a new blockchain that could be compared with the original one that is already sent to other nodes and for those reasons and others, this new modified block will be considered as an invalid block so it will not be accepted by the chain. The configuration updates period which is 10 seconds will force us to use the simplest aspect of blockchain cryptography in our proposed framework for that it's a short period of time which means that the difficulty will have to be not so high but effective at the same time.

It is worth mentioning that the difficulty will not be so high because our goal is to protect the network information from corruption and make the data tamper-proof not create highly efficient cryptocurrencies here; since, higher difficulty means more load on the network and higher consumption of its resources and that will make the network more DoS/DDoS attacks - prone which will lead back to where the whole issue originally started; because it is needed to make a balance between our needs of security and the load on the network nodes. So, it's possible to say that a puzzle will be made, that node/nodes of the network, in this case the controllers will have to find its solution in order to validate one block in the chain which will contain the data of network configuration and that will be done to each block, which means that each update of configuration will be a block in the blockchain of update flowing through the network structure.

The puzzle is done by including the configuration information into the blockchains and that's by adding a number to the data; so that the hash number resulting from mixing both the number and the data together will have a specific range of zeros and that is the difficulty that can be chosen as needed in the network's blockchain and that will make the node try guessing the right number by choosing different numbers randomly till finding the right number which won't be so hard or so easy but it won't take long for the reasons mentioned before but, it will be in 10 seconds after the issuing of an update; this number is called a "nonce", which is a concatenation of "number used once." In the case of bitcoin, the nonce is an integer between 0 and 4,294,967,296 [102]; that kind of guessing will be the proof of work-like for our framework and that will make it hard to tamper with the data because if the attacker tried to add some blocks of data of his own, these blocks will have to have the right required number to be added to the data of network and passed to the hash function to create a specific hash number with starting specific amount of zeros and that's the real puzzle for network's node and to make his block contain the hash of the previous block, all that has to be done in order for him to be able to add a valid block of data and tamper with the information stored in the chain; it will also require the perpetrator more time and create more obstacles in his way in order to manipulate the data mean while our framework and the already built-in techniques will detect his actions so fast maybe before he can

even start guessing. And that would prove the ability to use blockchain technology to assure the security of the data flowing and broadcast messages of software-defined network. Finally, it's possible to use the Marconi protocol [103] to achieve this idea. It is worth mentioning that Marconi protocol is new protocol specialized to help securing Ethernet-based networks using some blockchain-based technology by targeting layer 2 of the open system interconnection OSI model of networks which is the datalink layer and creating programmable packets, it allows developers to create and deploy intelligent, decentralized networking applications that can be run by nodes or end users. Blockchain projects, private institutions, and enterprises can utilize the network and the platform it's built on to manage their infrastructure and develop smart distributed networking and cybersecurity services and also as an Example of applications that could be developed by Marconi aside from software-defined networking (SDN); there is intrusion detection and prevention systems (IDS/IPS), anti-malware and anti-virus protection, content delivery networks (CDN), virtual private networks (VPN), and new blockchain protocols [104]. But there are some differences between our proposed blockchain-based solution and the Marconi protocol for instance; our framework works on transport layer security TLS layer, since that it uses the RSA asymmetric cryptography algorithm and openflow protocol that connects the controller with the switch, are both working on that layer. also, our proposed framework operates on the network layer since it uses the IPsec algorithm that is the basis of VPN technology which is used by the network layer while, Marconi uses datalink as its work environment so, our framework targets different layers and even if it might somehow work on datalink then, it will not be the only layer that it targets. Also, Marconi protocol uses smart contracts for defining how long the exchange of data will be and how much data can be exchanged, and at what price of fuel while ours, could use smart contracts only for authentication of the connecting parties. Also, Individuals, network operators, and internet service providers can participate by contributing their bandwidth or compute resources to the network; in return for contributing resources and processing network traffic, nodes periodically receive network tokens known as Marcos. The marco is the base unit of measurement for distributed networking and computing, the fuel consumed for network usage, administration, and smart contract processing [104]; while our blockchain solution doesn't do that, it is merely using an aspect of the blockchain technology and leveraging its cryptographic ability in a light weight for securing or assuring the security of the information exchanged between the controlling nodes in the control plane of SDN.

End users can utilize the Marconi network to access the internet or nearby compute power, either by procuring Marcos or by mining them through operating a contributing node,

while ours doesn't have that and it doesn't have that kind of complexity, these differences and others are differentiating our proposed blockchain-based approach from the Marconi protocol.
In the future, it's possible to develop this method into a new approach by sharing the blockchain across the whole network nodes including PCs, switches, routers etc.

Our framework app could be developed in the future to serve deeper purposes and to have higher and better features, it's even possible to make the time period of the configuration updates shorter or longer as per our needs. Also, in the future some issues might emerge and have to be addressed like the running out of numbers guessed to create valid blocks which means running out of blocks and that could be after a really long time; just like what will happen to bitcoin cryptocurrency in the future, where it is estimated that mining the available 21 million bitcoins will reach its maximum peak of course by the year 2140 [102] so, getting back to our situation if this case will be faced by the network administrator then it is simply possible to tweak some features in the blockchain created and distributed among the controllers of the software-defined network, one of the features that could be changed is the difficulty which means that the nodes will create a totally new blockchain containing the rest of configuration updates. And that could mean restarting the app and getting back to work but, it won't take a while and if needed it will be done after a long time as mentioned before.

Defense against attack on serial topology. The three controllers working in the topology will be sharing the blocks of information with their hashes that originally originate from the main controller which is one of the three nodes, in case of an attack on the main controller and if the perpetrator tried to change the information or data of updates broadcasted to the backup controllers then, he'll have to change their hashes and to change their hashes; he'll have to know and change the hash of the previous block, because those controllers will hash the coming blocks again and compare them with the original hashes escorting the coming blocks and as mentioned before any slight change of data inside the block will change the hash of that block also, the backup controller which is the receiving node of the configuration update will compare the coming or received hash with the hashes it already received previously and figure out the difference which will cause the dropping and rejection of the manipulated hash. Of course, since the update will occur every 10 seconds then the hashes will be sent every 10 seconds.

Defense against attack on parallel topology. Here there are three controllers working as a whole entity as mentioned before and since they are equal peers for each other, the controllers here will broadcast to each other any change they have in their information and update each other's tables with network configuration every 10 seconds as well, every one of them will hash the coming blocks again and compare them with the original hashes escorting the coming blocks

and also will compare the hashes of information they have with coming new hashes of the new information and check if they have the hashes of the previous blocks and if they match and of course if they don't match then they'll drop them and declare that sending controller as an infected controller and start taking the right measurements.

Defense against attack on hybrid topology. Here the updates will be between the main and the backup controllers but, since it has also a feature of parallel topology which is three main controllers working together as a whole so, it's possible to add hashed updates between the main controllers as well because we'll need them to check each other's information. Every controlling node will hash the received blocks again and compare them with the original hashes escorting the received blocks, and also will compare them with hashes of previous blocks and if they don't match then they'll be rejected. Of course, in all the aforementioned topologies, the received block will be dropped in case if its hashes don't match and that by nature will alert the Hydra framework about an attack.

It is possible to see now how each one of those algorithms can help in the assurance of security of software-defined networks hence, assuring the security of the networks in general. In a whole the general form of the suite of algorithms applied in the framework will be like what is shown in the figure 2.11 below.

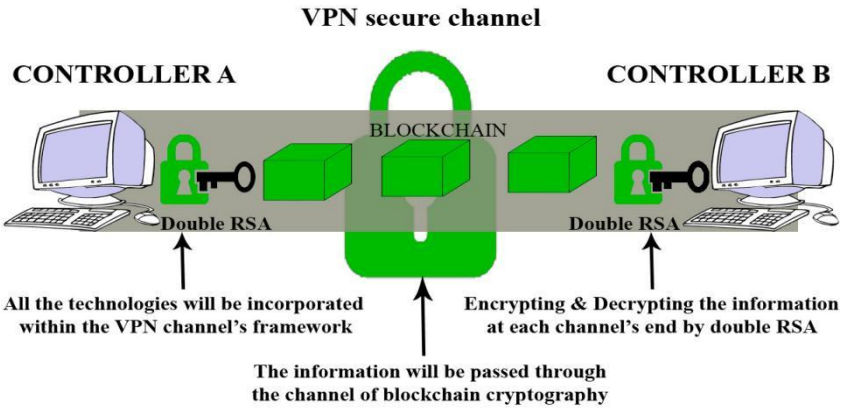


Figure 2.11. The architecture of Hydra suite

2.3 Topologies proposed for assuring the security of SDN

2.3.1 Serial Topology

As seen in the figure 2.12 below the topology contains 3 controllers, where there's a main controller and two backup ones just in case of an attack or a disruption that may stop the first or main controller. the main controller controls the whole network and its nodes and sends an

update every 10 seconds that informs the backup controllers about any change in the topology or network configuration and it also acts like a beacon that alerts the controllers if there's a latency in the update message and it took more than 10 seconds then, it will be taken that the main controller has been infected by a DDoS attack or any type of threat hence, comes the role for the second controller which was a backup to become the next main controller and the same thing now goes between the new main controller and the third controller which now becomes the new 1st backup controller until the previous main one will be maintained and restored then everything goes back to its previous state.

And of course, if the new main controller which was previously the second one got attacked as well then, the same procedure will be applied and it will be replaced with the third controller.

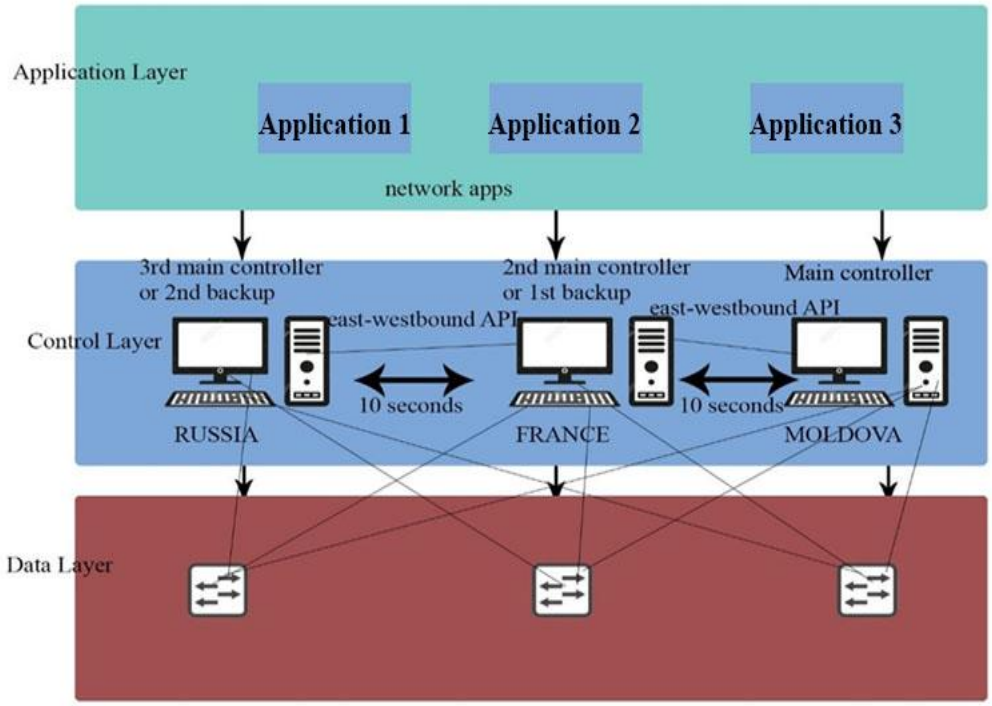


Figure 2.12. Serial Topology

Of course, the assignment of controllers could be by using a priority number to choose the main, the first backup (second) and the (second backup) third controller. And as mentioned above in the Hydra section, a botnet program will be added to all controllers so that they can install it in the computers connected to them for a counterattack measurement to disrupt the attacker's machine and prevent it from continuing its attack by creating a DoS/DDoS attack on it or by sending a simple virus to it that will stop it or force it to restart then the next procedure will be isolating or blocking the IP of that attacker's machine.

2.3.2 Parallel Topology

In this topology we'll use also 3 controllers as well but, they'll work together as a whole one entity integrated together where the info is processed synchronously and each controller will deal with any area or slice of the network especially, if it was closer to this slice or segment than other controllers; which means each controller will give higher priority to closer network segments than the further or more distant segments (of course the distance will be calculated based on the number of hops in the path between the network slice and the controller), of course the updates will be also every 10 seconds. This topology is named as the parallel topology since all controllers work together in parallel so, there is no priority numbers here because they all behave like parts of one main controller where each one of its nodes will serve the closest switches to it first so, the priority for a switch in France to be served by a controller in France will be higher than that of a switch in Russia based on the number of hops between the switch and the controller, as shown in the Figure 2.13.

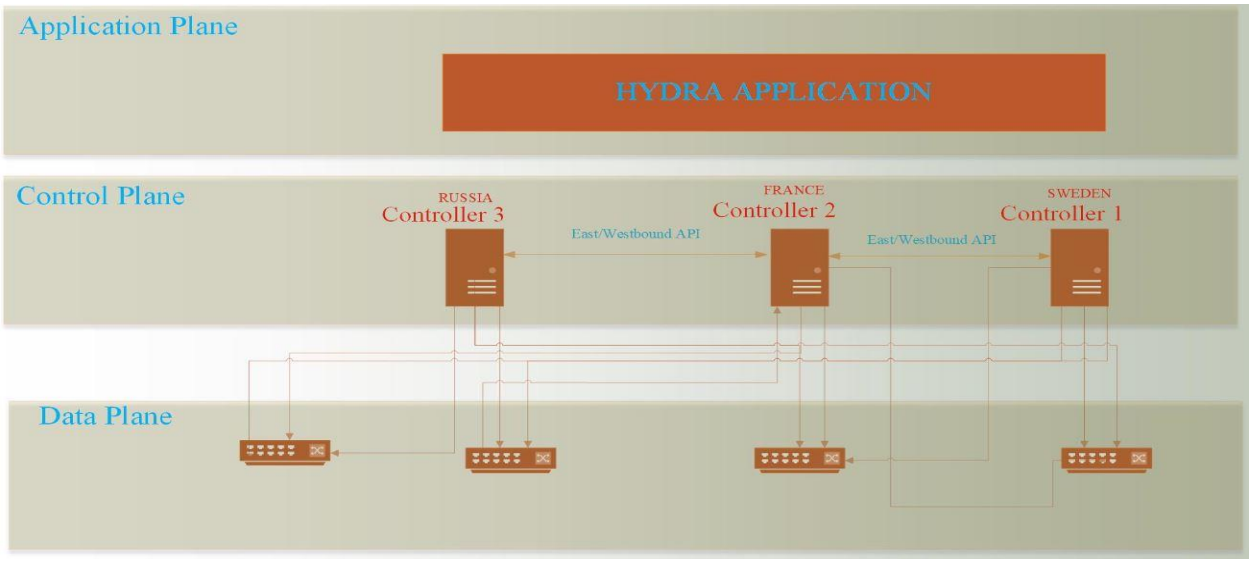


Figure 2.13. Parallel Topology

Which means that if there are three controllers connected in a parallel topology and they're distributed in France, Moldova and Russia respectively then, the part or node of the that big main controller that lies in Moldova will serve the switches lying in Moldova for speed reasons and those other switches lying in Russia and France will be served by the controllers of their own countries and in case if the controllers of their countries were busy and the Moldovan node was free then, it will automatically serve the Russian and French switches because all controllers are integrated with each other in a parallel way so, they behave as one distributed controller over those countries. Despite that the controllers here work simultaneously like one and share the same database of configuration but, the updates will be also every 10 seconds but

they will be mere acknowledgements. In case of a DoS/DDoS attack, it is possible to have the other controllers fill the place of the infected controller and try to deal with the coming flooding requests. In such situation, there will be one of two cases:

1. The infected controller will be isolated using access control lists or any other technology. And others will keep working like before like one controlling entity containing the rest of controllers.
2. Or the 2 other controllers will try to ease the pressure off of the infected controller by taking a huge load of the requests coming from the attacking source to prevent the attacked controller from being down completely. Until the attack source gets blocked and counter attacked; or if that fails and the other 2 controllers behave slowly or respond too late then, they will attack the attacking source with a counter DoS/DDoS attack and then they'll block both the attacker's IP and the infected controller's as well and we'll go back to step 1.

2.3.3 Hybrid Topology

As seen from the Figure 2.14, this topology combines features of both the previous topologies, meaning there are six controllers here. There will be three main controllers that work like one controller simultaneously and in a parallel way and each one will have a backup controller just in case if it's down then, the backup will take control instead of the infected one. The only update will be between each main controller and its backup one, and it will be every 10 seconds as well. Every backup controller will be connected to other backup controllers alongside with switches in the network and its own main controller that it assists as well. The priority numbers will be between every main controller and its backup one only.

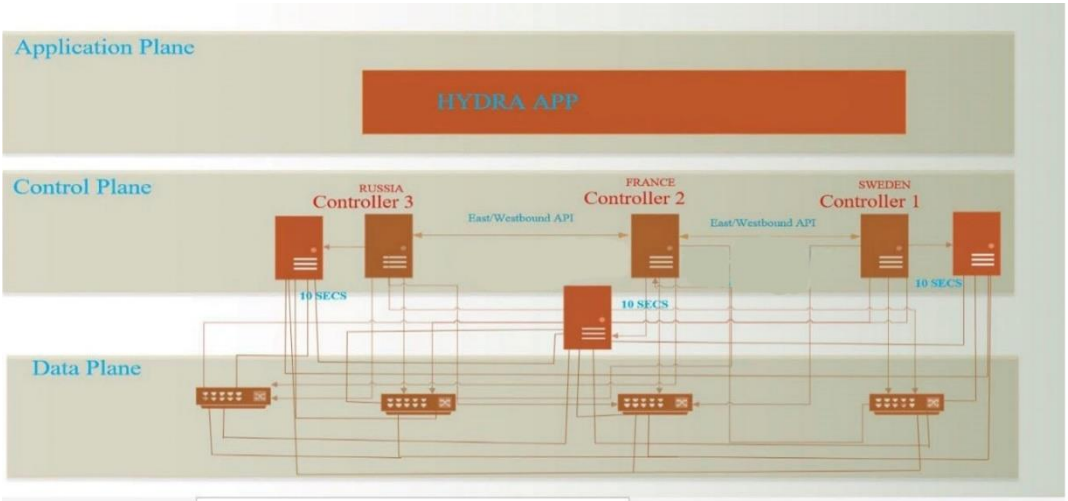


Figure 2.14. Hybrid Topology

The fourth topology which is the ordinary usual one containing a controller the behaves as the brain of the network; fails to achieve its role in the existence of these ever evolving threats and this topology fails to implement the proposed algorithms and topologies as well, because they are meant for multiple controllers' topologies.

2.3.4 Ordinary Topology

Here there is a basic topology of software-defined networks, where there is one controller that controls the whole network, it controls the switches and they control the rest of the network of course; here the controller will be serving the computers by serving the switches that transfer the requests of the computers. But, here if there are too many computers requesting to be served or a DoS/DDoS attack on the controller and that attack was able to disrupt the server/controller and there is no backup controller or parallel controller that works with our main controller hence, that will stop the controller with no substitute and it might jeopardize the whole network by stopping it or hacking into switches by controlling the controller itself or by giving the network commands to let unauthorized entities or devices and that will mean the end with no ability to recover. The figure 2.15 below shows an example of the ordinary one-controller topology.

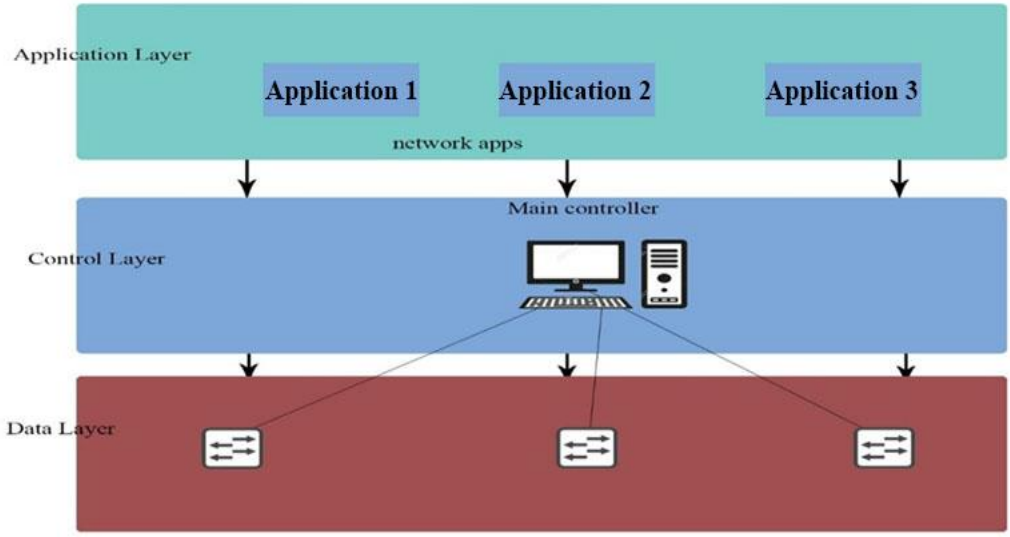


Figure 2.15. Ordinary Topology

2.4 Effect comparison and analysis of each proposed algorithm on each controllers' topology

As shown in Table 2.1, the algorithms proposed are integrated in a suite or a framework called the Hydra framework which is in turn has some DoS/DDoS attacks counter measurement algorithms and the whole framework integration with the suggested topologies is optional.

Meaning that it is possible after developing the framework software to activate/deactivate it as per the needs of the network administrator and the business requirements and in the aforementioned table 2.1 there is a brief comparison of the effect of each algorithm or methodology on each of the proposed SDN controllers' topology; alongside with the usual exiting one which is the ordinary topology.

Table 2.1. The Effect of Each Algorithm on Each Controllers' Topology

Topology	Algorithm			
	Hydr	**VPN**	**RSA**	**Blockchain**
Serial	Flexibility, ability to maintain itself.	Protection and encryption.	Encryption	CIA: confidentiality, integrity and authenticity.
Parallel	Doesn't work 100% due to that all controllers work together so in case of an attack it could disrupt them all which hinders implementation of Hydra.	Protection and encryption.	Encryption	CIA: confidentiality, integrity and authenticity.
Hybrid	Flexibility, ability to maintain itself and continue working even after attack.	Protection and encryption.	Encryption	CIA: confidentiality, integrity and authenticity.
Ordinary	None	None	Restricted in communication between the controller and switches.	None

2.5 Conclusions of Chapter 2

1. Different algorithms and techniques were described regarding the assurance of SDN security in precise and computer networks in general. It was proposed the usage of the existing Virtual Private Networks (VPN) algorithm between the SDN controllers of every topology, by using the IPsec technology to enhance the security of the east-west bound API connection of the SDN controllers against some attacks like eavesdropping and the Man In the Middle Attack (MITM). Based on the TCP/IP model, our framework mostly targets mostly DDoS protocol attacks since those attacks are mostly work with TCP, UDP packets using the SYN flood attack, those protocols are operating in the TLS layer. Since that most Hydra protocols like Ipsec of the VPN algorithm are working in the TLS layer plus the already existing SDN OpenFlow protocol which is working in TLS as well. Then, its's possible to say that the proposed framework deals mostly with protocol-based DDoS attacks that are basically targeting the TLS layer.

2. It was newly presented the usage of a modified version of RSA algorithm which was named as the Double RSA algorithm in this research to enhance the RSA cryptographic effect between every two controlling nodes in the SDN topology. Also, Blockchain technology was proposed here in a new modified way to carry the hashed version of the network information,

configurations and requests for verification purposes. All those previously proposed algorithms were incorporated within a newly created framework named the Hydra that has other features aside from the previously mentioned algorithms like, the DoS/DDoS counter attack measurement principle to counter attack the attacker using the networks' available nodes.

3. They permit to analyze the effect of the proposed algorithms that shows their efficiency in assuring the security of SDN and consolidate its defense against cyber-attacks like DoS/DDoS attacks. these algorithms solve those issues:

   a. Centralization: despite that SDN controllers give the ability to manage the network environment from a single point but that also could be considered as a weakness point if jeopardized and used as a single point of failure.

   b. East-west bound API Connection: for that issue, it was proposed the usage of the Hydra framework that contains different algorithms.

   c. Security level measurement: to solve that; the research provided different mathematical parameters to assess the security level SDN environment.

4. There were proposed the new topologies of SDN controllers which are the Serial topology, the Parallel topology, the Hybrid topology and since there was a need for a reference point for comparison of the proposed topologies performance, then it was imperative to describe a single-controller topology that was called the Ordinary topology in this research.

5. The importance of the proposed topic revolves about assuring the security of software-defined network and making it even a better and a safer environment for a faster and a more agile transition of networks from the classical structure of network management to the SDN structure.

6. The novelty of the proposed framework stems from the usage of new and existing methods and algorithms in modified new way and some of them are not even used in the SDN at all and in this research, their usage is proposed in a unique way to serve as defense shields against different threats and attacks like man in the middle (MITM) and Denial of Service/Distributed Denial of Service (DoS/DDoS).

3 EFFICIENCY EVALUATION OF THE PROPOSED TOPOLOGIES FOR COMPUTER NETWORKS SECURITY ASSURING

3.1 Defining the problem

It is necessary to evaluate the topologies proposed in chapter two regarding the efficiency of security protection, the analysis of the literature showed that to the best of our knowledge there are no existing methods and approaches for evaluation of security protection of the SDN-based networks.

Taking into account this fact in chapter three, new methods were proposed for evaluation of security level protection of SDN-based computer networks. The proposed methods were applied for elaborated SDN topologies' controllers.

This chapter reviews some SDN controllers that could be used with this proposed framework despite that it could work with all other SDN controllers. Then it focuses on the mathematical theoretical and practical side of the thesis by modeling the presented decentralized topologies solely in Petri Nets methodology in the normal condition and under a DoS/DDoS attack, then compare between them on one hand and between them and the classical exiting one controller-model on the other hand to derive a set of theoretical security risk assessment parameters that later could be proven practically by using the results gained from the Petri Nets models in the equation itself.

3.2 Overview of the existing methods and applications for evaluation of computer networks security level

There are various proposed security level evaluation methods for computer networks in general, depending on the aspects of security needed to be measured, platforms, approaches and types of networks and their connections and many other elements that could be considered, all that would create diverse methods and types of laws to determine the security risk that affects a computer network. For instance, the work using Petri Nets to map the SDN environment [105], where their objective is to analyze the security via the transformation state of communication model. They provide the analysis method of security based on token to determine the potential threats. Also, they present analysis for the SDN via the combination of the number of token and time series based on Petri Net. So, it is possible to say that they used PN system mostly for analysis of the security aspect of SDN. Another research was [106] where the researchers use elliptic curve equation to create a unique signature that will be used to verify data packets entering the SDN controller. Also, research from China [107] has leveraged a combination

model to evaluate the computer network security to improve the evaluation accuracy of computer network security. The combination model used particle swarm optimization (PSO) to optimize the parameters of back propagation neural network (BPNN). While some other works focus on mobile networks [108] where the researchers in this paper, have developed a performance evaluation mathematical model for firewall system of mobile networks and that's by leveraging queuing theory for a multi-hierarchy firewall with multiple concurrent services. Also, the researchers have employed the throughput and the package loss rate as performance evaluation indicators. They conducted discrete-event simulated experiments for further verification. Experimental results were compared to theoretically obtained values to determine a resource allocation scheme that could present optimal firewall performance and can offer a better quality of service (QoS) in mobile networks. By applying an Erlang queuing model, this work suggests a two-phase multiservice station and multiprotocol firewall model with multiple concurrent applications. So, it is possible to conclude that in this research, based on different protocols, phases and applications, the values of performance indicators like packet loss rate and throughput were obtained. Not to forget that an optimal solution was derived after comparing simulation results and theoretical computation and there has been an enumeration of combinations of resource allocations by using twelve resources. This research also compared the error between the simulated experiment values and the theoretical computation values, proving that the presented model can exactly represent the firewall handling process. Last but not least, the mathematical modeling made for SDN [109], this work proposes a novel approach of anti-fragile cyber security within SDN structure, it also proposes a unified model for the integration of both approaches of "Security with SDN" and "Security for SDN" to implement the overall objective of protecting information against cyber threats in the globally connected internetwork. So, to the best of our knowledge there aren't researches that permit the effective evaluation of the security level assurance of the elaborated SDN controllers' topologies.

Taking into account this fact, a new approach is suggested to evaluate the security level efficiency of SDN controllers' topologies based on Petri Nets (PN) applications and a set of parameters which are the Reliability of Service (RoS), Defense Factor (DF), Risk Factor (RF), Modified Risk assessment (RM) and last but not least the Cost effectiveness of every suggested topology as compared to the Ordinary topology.

3.3 Applications of Petri Nets for efficiency evaluation of security level of computer networks

Petri Nets field was invented by Carl Adam Petri for the purpose of describing chemical processes. A Petri net, which is called as a place/transition (PT) net as well. It is described as one of many available mathematical modeling techniques used for the purpose of modeling distributed systems. Also, it could be described as a discrete event dynamic system. The petri net is a directed bipartite graph, meaning that it contains mainly of two types of nodes which are places (i.e. conditions, represented by circles) and transitions (i.e. events that may occur, represented by bars).

The directed arcs or arrows describe the direction of the procedure meaning which places are pre- or post-conditions for which transitions. Petri Nets technique offers a graphical notation for stepwise procedures or processes that could include iteration, concurrent execution and/or choice. This technique has an exact mathematical definition [110]. The places in this system could have a discrete number of marks named tokens. A marking is any tokens' distribution over the places which will mean a configuration of the net. In other words, in the diagram of Petri Nets, a transition of a Petri net could fire if it was activated or as it is called enabled, meaning that if there were enough tokens in all of its input places; the time the transition fires, then it depletes the input tokens, and issues tokens in its output places. The firing procedure is atomic, which means that is a single non-interruptible step.

Unless an execution configuration is identified, the execution of Petri nets is nondeterministic meaning, if multiple transitions were activated or enabled at the same time, then the transitions will fire in any order.

Petri nets are suitable for the concurrent behavior of distributed systems modeling, because the firing procedure is nondeterministic, not to forget that multiple tokens could exists anywhere in the diagram.

Petri Nets system has also different types like colored Petri Nets which are used for various purposes [111] where the researchers in this work have used colored fuzzy Petri Nets to model and simulate membrane systems which are enriched by fuzzy kinetic parameters. They also presented a methodology and workflow by leveraging colored fuzzy Petri Nets to model and simulate general biological systems that have to cope with incomplete knowledge of their kinetic data.

By using the modeling feature the Petri Nets system has, it is possible to have a better understanding of the problem and its nature to get to the next phase of the implementation of Petri Nets which is, the simulation process. After the simulation it is possible to conclude, gain

numerical results that will be used in other experiments or to derive mathematical relationships based on the gained numerical results. One of the usage attempts of Petri Nets system to model the SDN environment was the Chinese researchers in [105] where they analyzed the security by the communication model's transformation state. Also, they provided a method for security analysis based on token to figure out potential threats. They analyzed the SDN through a number combination of token and time series based on Petri Nets system, and gained the results.

There are different types of simulation software that use petri net models so, one of the best choices was selected, which is the platform independent Petri Nets editor (PIPE) software and one of its main modules is used for this research which is Generalized Stochastic Petri Nets GSPN analysis module which concentrates mainly on the task ahead which is reviewing the number of tokens occupying the places that represent the controllers of SDN.

Here it will be demonstrated, how the three topologies of controllers work and interact with each other and against a DoS/DDoS attacks using Petri Nets which use Markov chains for modeling the behavior of the controllers in each topology. We will use the PIPE software to implement our topologies in Petri Nets, especially the Generalized Stochastic Petri Nets GSPN module to get results. A GSPN is a 6-tuple (P, T, F, W, M_0, λ) module where:

1. $P= \{P_1, P_2 \ldots P_m\}$ represents a set of finite nature of places, where $n \geq 0$.
2. $T= T_1 \cup T_2$, $T1= \{t_1, t_2 \ldots t_m\}$ represents a set of finite nature of timed transitions, and every transition is correlated with a random time of delay between the functions of enabling and firing. Also, $T2= \{t_{m+1}, t_{m+2} \ldots t_n\}$ represents the set of immediate transitions of a finite nature. It is possible to fire these transitions randomly. In addition, the delay time will be zero.
3. While F can be described mathematically as $F \subseteq (P \times T) \cap (T \times P)$, which is an arc set.
4. W refers to the weight arcs operation, so: F $\rightarrow$ {1, 2, 3…}.
5. In addition, $M_0: P \rightarrow \{0, 1, 2, 3 \ldots\}$ represents the initial marking, that can be described as $(P \times T) = \emptyset \cap (T \times P) = \emptyset$.
6. Finally, $\lambda= \{\lambda_1, \lambda_2, \lambda_3 \ldots \lambda_n\}$ describes firing rates group or category referring to timed transitions. Where every single rate represents the transition's average firing times per unit of time [112].

Our goal now is to determine the best topology or the most suitable one for the best SDN environment's performance with more reliability and ability to deter cyber-attacks like DoS/DDoS attacks.

At present the theory of Petri Nets (PN) is well developed for different applications [113]. The theory previously was used for modeling SDN-based technologies mostly and wasn't used for evaluation of security level assessment of computer networks based on SDN technologies as

we propose in this research. We propose to apply the PN theory for evaluation of security level protection in next mode.

3.4 The new method of the computer networks security level evaluation based on the Petri Nets and a set of parameters

In the first stage for topology of SDN controllers to be investigated, a Petri Nets model is elaborated using theory described in section 3.2. For example, for topology presented as the figure 2.12 and its elaborated Petri Nets model that is presented in Figure 3.1.

The next stage, the simulation of the elaborated model is performed using the Generalized Stochastic Petri Nets (GSPN) module which is a 6-tuple (P, T, F, W, M_0, λ) module where, $P = \{P_1, P_2 \ldots P_m\}$ is a finite set of places, $n \geq 0$, and obtaining the numerical data regarding the places and tokens. A sample of the data acquired from the modeling of the Serial topology from figure 2.13 can be presented in Table 3.1.

Table 3.1. Example of average number (distribution intensity) of tokens in controllers in SDN topology

No.	Places	Topology	
		No. of controllers	Tokens Z_{Ki}
1	$P3$	1	0.16
2	$P7$	1	0.06
3	$P11$	1	0.13

At the third stage, for obtained data of the simulation we propose to apply a set of parameters, to evaluate the security level of computer networks. The proposed parameters are Reliability of Service (RoS), Defense Factor (DF), Risk Factor (RF), modified security Risk assessment law (RM) and Cost effect (Y).

Reliability of Service (RoS).

Is an expression added in this research to describe the performance reliability of data protection. Reliability of Service (RoS) could be close to Quality of Service (QoS) which is the definition of the performance of any system's service, like a computer network or a cloud computing etc. The SDN has many positive features, as mentioned previously in Section 1.2. These features have attracted the attention of researchers to improve the QoS provisioning of today's various network applications [114]. But RoS is a little bit different and has a more detailed specification tailored for the needs of SDN. We propose to measure the RoS as follows:

$$RoS = 1 - TANR / TANR_{O.} \quad (3.1)$$

Where *TANR* is the Total Average Number of network requests of the proposed topologies, which is the summation of all averages after dividing them by the count of these averages and that means how many requests per server and shows the risk occurrence level. The $TANR_O$ is the Total Average Number or intensity of requests in controller of ordinary topology.

Defense Factor (DF).

To determine the strength and security feasibility of the network against DoS/DDoS attacks. In terms of Petri Nets, the requests will be represented by how many tokens are there in specific places which in turn represent specific nodes in the software-defined network and those specific nodes of interest are the SDN controllers. In the equation (3.4) places representing the SDN controllers are denoted as K, where $K \in P$, P is the whole group of places in the Petri Nets (PN) model, which in turn is a tuple of 5 objects, $PN = \{P, T, I, O, M_0\}$, where P is the finite set of places, T is a finite set of transitions, I is the input function, O for output function and M_0 is the initial marking.

$$K = \sum_{i=1}^{i=n} Ki,\ K_i \in \{1 - n\}, \tag{3.2}$$

$$Z = \sum_{i=1}^{i=n} Zi,\quad Z_i \in \{0 - \infty\}, \tag{3.3}$$

$$DF = [\sum_{i=1}^{i=n} Ki] / [\sum_{i=1}^{i=n} Zi] = \sum_{i=1}^{i=n} [\frac{Ki}{Zi}]. \tag{3.4}$$

Where K is the number of the places that represent the controllers in a specific model, K_i= = (K_1, $K_2 \ldots K_n$) and Z is the value of tokens (requests) in those places K_i, Z_i = ($0 \ldots \infty$).

The Defense Factor *DF* equation depends mainly on the assessment of the strength of software-defined network's controllers based on their emptiness and that means their readiness and availability to deter any kind of DoS/DDoS attack. Therefore, it is logical to say that the more controllers we have the better the network's capability it is to deter those attacks. That means the more controllers the better and the higher DF value it is and that explains why the DF equation has the places that represent the controllers in numerator position because the number of controllers is proportional to the value of the DF.

While on the other hand we can see that the more tokens that represent the requests, the weaker the network gets and that means the less DF value we will get.

So, the DF value and the number of tokens or requests are inversely proportional and that's why they should be put in the denominator position. In addition, it is mentioning worthy that if we reversed the DF then, meaning if we divided the tokens/requests by the places/controllers then we'll get how many requests per server meaning; it will divide them equally and that could be unrealistic, because based on the types of reactions between the

controllers in every topology; a controller could be dealing with more requests while the other is just waiting as a backup controller.

Risk Factor (RF).

To figure out the weakness level of the network environment. The values of the Total Average Number (distribution intensity) of Requests (TANR) in section 3.3.1, can be described as the Risk Factor. Since we need to find the strength and defense ability of the network controllers to deter the DoS/DDoS attacks and the more tokens/ requests we have, the more occupied the controllers will be and the weaker they will be and this way we will not get defense ability or reliability level of the network but rather the weakness point.

So, we propose also the Risk Factor (*RF*) parameter to evaluate the efficiency of the elaborated topologies. The Parameter can be estimated as:

$$RF = 1/DF. \tag{3.5}$$

Where *DF*, is the Defense Factor value of the measured topology. The parameter *RF* will still be the opposite of what we need to figure out, which is the *TANR* also. So, the places / controllers should be in the numerator and the tokens/requests should be in the denominator and that is another logical proof of why there was a need to derive that formula and to put it in that form.

Modified security Risk assessment (RM).

To assess the security level of computer networks, it's possible to leverage the risk assessment law but to make it suitable for SDN then it's possible to modify the risk assessment law and after modification, it's called in this research, the modified risk assessment law (*RM*). Based on the modified law of security risk assessment (*RM*), we can apply the security risk assessment law [115] to evaluate the protection level of the computer networks, which states:

$$R = P_0 * V. \tag{3.6}$$

Where *R* is the security risk assessment that quantifies and shows the possibility of a threat acting upon a vulnerability successfully and the severity of the results of that attack, P_0 represents the initial probability or likelihood of the vulnerability occurrence and *V* represents the value or cost of the asset.

In other words, using this formula we can estimate how much our proposed framework will reduce the security risk of a computer network, hence assuring its security. Since a server is the most important part and has the highest value node in the network environment, in which we will install our controller software, then we can say it has the highest asset value or impact. The servers have the value *V*=100 as an asset impact [115] because it is the value of the server's

impact on the secure socket layer (SSL) which is the same layer in which the OpenFlow protocol works.

The probability of vulnerability P_0 = 0.025 which is measured based on the lost orders due to the web server denial of service attack. We can gain a new value of probability P_n, which will be affected by the DF mathematically as shown:

$$P_n = P_0 / DF. \quad (3.7)$$

The higher Defense Factor, the better which is the opposite of the probability of vulnerability; that means that they should be inversely proportional mathematically as they're logically and that's why they're positioned this way in the formula of finding the new likelihood or probability of vulnerability.

It is possible to estimate that our framework and the formula derived from its modeling, will reduce the likelihood of attacks occurrence (like DoS/DDoS attacks) and for that we propose a modified version of Security Risk Assessment parameter and it will be as:

$$RM = P_n * V. \quad (3.8)$$

The usage circumstances of these parameters differ from each other. When it is needed to measure the data protection performance then, it is possible to apply the RoS parameter. When there's a need to determine the defense ability and the network's strength against DoS/DDoS attacks then, it is possible to depend on the *DF* parameter. While if there was a need to reverse the situation and figure out the weakness of the network then, it is possible to reverse the parameter and use the *RF* which is the inversely proportional to *DF*. In case there was a need to measure the SDN environment weakness based on the computer networks law, then it is possible to modify it to be used based on the proposed SDN environments and their Petri Nets modeling and in this case the *RM* parameter can be used.

Cost effect parameter (Y).

To evaluate the influence of the costs of using the proposed SDN controller topologies for SDN security assurance, we used the data regarding the prices of the switches, computers and controllers from [116], [117].

In chapter 2, sections 2.2.1, 2.2.2, 2.2.3 and 2.2.4, there were described the basic structures of different topologies - Serial, Parallel, Hybrid and Ordinary respectively. Now it's time to estimate the cost of those topologies, taking into account as an example that Ordinary topology will include 1 controller, 3 switches and 6 computers, both the Serial and Parallel topologies will have 3 controllers, 3 switches and 6 computers and the Hybrid topology will contain 6 controllers, 3 switches and 6 computers.

In this case, the total cost *TC* of the Ordinary network topology can be estimated as:

$$TC_O = CC+3SC+6NC. \quad (3.9)$$

Where, *CC*, *SC* and *NC* are the cost of controller, switch and network computer respectively. The total cost of the Serial and Parallel topologies will be as follows:

$$TCS, P = 3CC+3SC+6NC. \quad (3.10)$$

And the total cost of the Hybrid topology will be:

$$TC_H = 6CC+3SC+6NC. \quad (3.11)$$

Let's establish the relations between the costs of the controller, switch, and computer as follows:

$$CC= aSC = bNC. \quad (3.12)$$

Where parameters $a = \{a_{min} \div a_{max}\}$, *and* $b = \{b_{min} \div b_{max}\}$. If we apply those values in the equations (3.9-3.11) then, the costs of the topologies could be described as:

$$TC_O = CC+3SC+6NC = CC\ (1+3/a+6/b), \quad (3.13)$$

$$TC_{S,P} = 3CC+3SC+6NC = 3CC\ (1+1/a+2/b), \quad (3.14)$$

$$TC_H = 6CC+3SC+6NC = CC\ (6+3/a+6/b). \quad (3.15)$$

In summary, the effectiveness of cost difference in using the proposed topologies can be shown by the following formulas:

$$Y1 = TC_{S,P}/TC_O = 3(2a+b+ab)/(6a+3b+ab). \quad (3.16)$$

$$Y2 = TC_H/TC_O = 3(2a+b+2ab)/(6a+3b+ab). \quad (3.17)$$

$$Y3 = TC_H/TC_{S,P} = (2a+b+2ab)/(2a+b+ab). \quad (3.18)$$

A sample of those five parameters will be in Table 3.2 which are taken from the Serial topology.

Table 3.2. Example of parameters of SDN topology

Parameter	*RoS*	*DF*	*RF*	*RM*	Value of *Y*
Data	0.94	8.23	0.12	0.3	1.63

In the last stage comes the analysis and interpretation of the obtained results.

3.5 Petri Nets modeling of the proposed SDN topologies

This section presents the modeling results of the proposed SDN topologies using Petri Nets which are Serial topology, Parallel topology, Hybrid topology and Ordinary topology.

3.5.1 Serial topology

As we can see in the Figure 3.1 we have 3 controllers working as one main controller and two backup controllers of course regardless of whether they were software-based or hardware-based controllers.

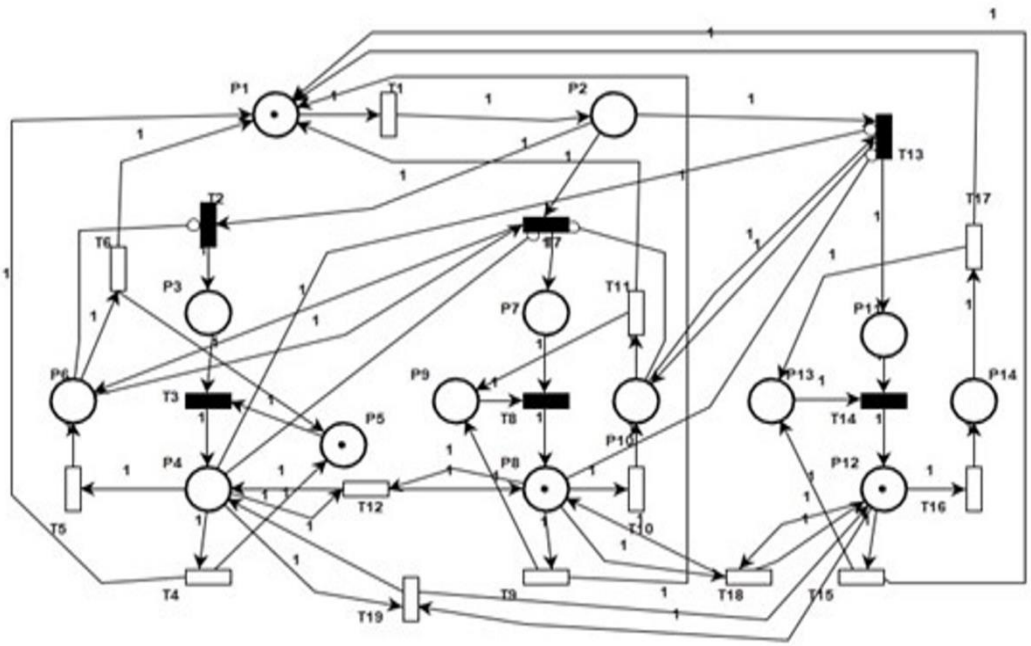


Figure 3.1. Serial topology modeling using Petri Nets

Description of the Serial topology scheme.

The main controller will be working normally as the only main brain unit or entity that manages the whole network behavior, interacting with switches and managing the requests of hosts through them.

1. The main controller will be sending updates of network configuration every 10 seconds to both backup controllers so that they be up to date and aware of overall network behavior, status and structure and be able to take control of the network in case of an attack or any kind of disruption of the main controller.

2. As mentioned before in case of an attack that will disrupt the main controller, the control and management of the whole network will be handed over to the next backup controller with the next closest priority number, in this case controller No 2.

3. A bot will be sent to infect the attacking source and then the attacker's IP will be blocked in one direction like what happens with the access control list so that we can still control it but it cannot reach our network, and the infected controller will be blocked and isolated as well.

4. The new main controller which was previously the backup one now manages the network and sends information updates of network configuration to the remaining backup controller till the maintenance of the previously main controller finishes. The Table 3.3 shows the description of places.

Table 3.3. Description of Places

Place	Description
*P*1	Packets processing by main server or controller/ the input place that sends data tokens
*P*2	Selection of servers
*P*3/ *P*7/ *P*11	Allocation of servers
*P*4/ *P*8/ *P*12	Server 1,2,3 Active processing
*P*5/ *P*9/ *P*13	Server 1,2,3 Free controllers
*P*6/ *P*10/ *P*14	Attack

While the Table 3.4 shows the transitions descriptions.

Table 3.4. Description of Transitions

Transition	Description
*T*1	Generated task i.e., packets processing
*T*2/ *T*7/ *T*13	Selection of servers 1, 2, 3
*T*3/ *T*8/ *T*14	Allocation of servers
*T*4/ *T*9/ *T*15	Task processing
*T*5/ *T*10/ *T*16	Exiting the stage
*T*6/ *T*11/ *T*17	Restoring the controller
*T*12/ *T*18/ *T*19	Updating the information and going back to initial stage of controllers

3.5.2 Parallel topology

This topology contains three main controllers that work together simultaneously as one entity like one brain and they update each other with the information they have about the contiguous network nodes each controller has. As shown in the figure 3.2.

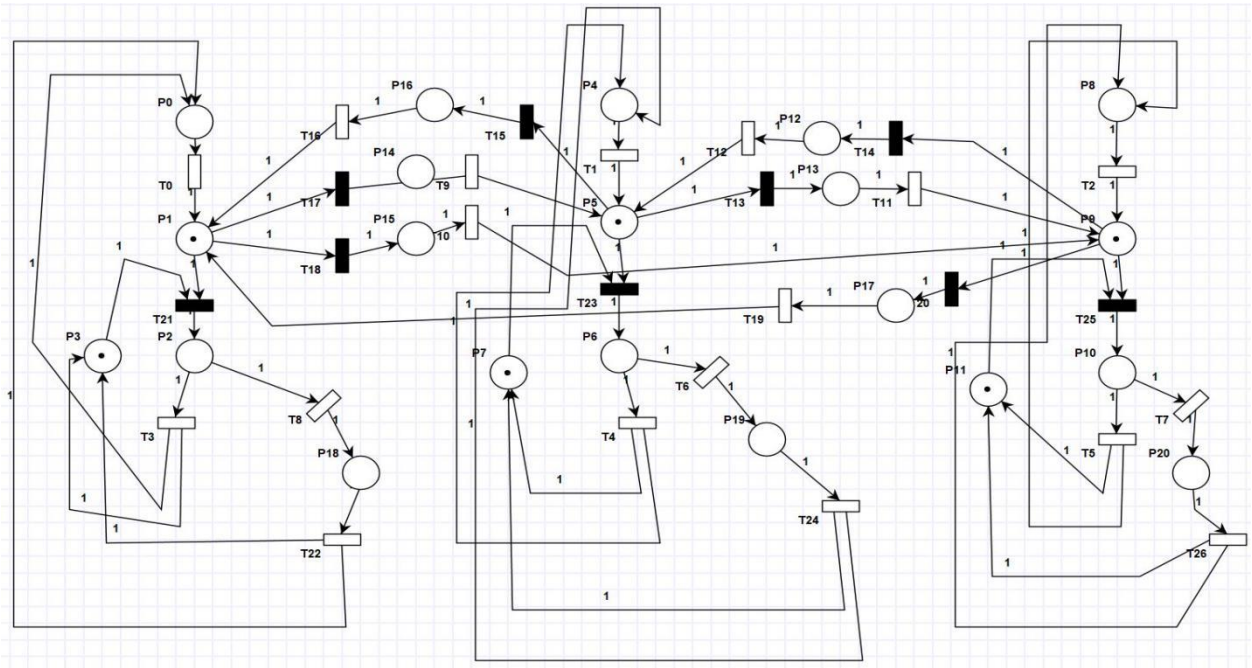


Figure 3.2. Parallel topology modeling using Petri Nets

Description of the Parallel topology scheme.

1. As mentioned before in this topology we'll have 3 controllers as well but, the difference will be in their interaction with each other.
2. The 3 controllers will be working as a whole entity like one brain of 3 parts where they work simultaneously to process switches' requests.
3. They will all behave like the main controller.
4. Each controller will serve switches and prioritize them based on the closest ones to it.
5. Each one of them will send a broadcast update to other remaining servers/controllers of its configuration every 10 seconds as well.
6. Since all the 3 controllers represent a whole one main controller and the updates between the controlling entities will be every 10 seconds, that means that the configuration information will be merged together every 10 seconds, in other words the main triple-parts controller will update its general of the whole network's status every 10 seconds.
7. In case of an attack on one controller, the other 2 controllers will fill the empty place of the infected one by serving the switches that were relying on the infected controller before it was infected or disrupted.
8. The switches will be already connected to all 3 controllers so, in case of an attack a bot will be sent to the attacking source and the infected controller's IP will be isolated alongside with the attacker's IP and all switches will be served directly without any noticeable change because the remaining 2 controllers will be able to add more space to deter a Dos/DDoS attack first and if that fails then the attacked controller will be isolated and the controllers continue their work like one controller of 2 main parts. The description of places is stated in the Table 3.5.

Table 3.5. Description of Places

Place	Description
*P*1/ *P*5/ *P*9	Allocation of servers
*P*2/ *P*6/ *P*10	Server 1,2,3 Active processing
*P*12- *P*17	Sharing the information and updating the network configuration
*P*3/ *P*7/ *P*11	Recovery and restoring the working state of the controller
*P*18/ *P*19/ *P*20	Attack on the server or controller
*P*0/ *P*4/ *P*8	Getting back to the initial state

The Table 3.6 below states the description of transitions.

Table 3.6. Description of Transitions

Transition	Description
*T*21/ *T*23/ *T*25	Server 1,2,3 Active processing
*T*9- *T*20	Sharing and updating the network configuration between servers
*T*3/ *T*4/ *T*5	Back to active processing/ processing the next request
*T*6/ *T*7/ *T*8	Attack on server/controller
*T*22/ *T*24/ *T*26	Recovery from the attack
*T*0/ *T*1/ *T*2	Transition back to initial state

3.5.3 Hybrid topology

Here, the topology consists of six controllers; three main controllers and three backup ones, where each main controller has one backup controller to be used as the main controller in case of a disruption, termination or any kind of attack on that main controller. The Figure 3.3 below shows the Hybrid topology modeling.

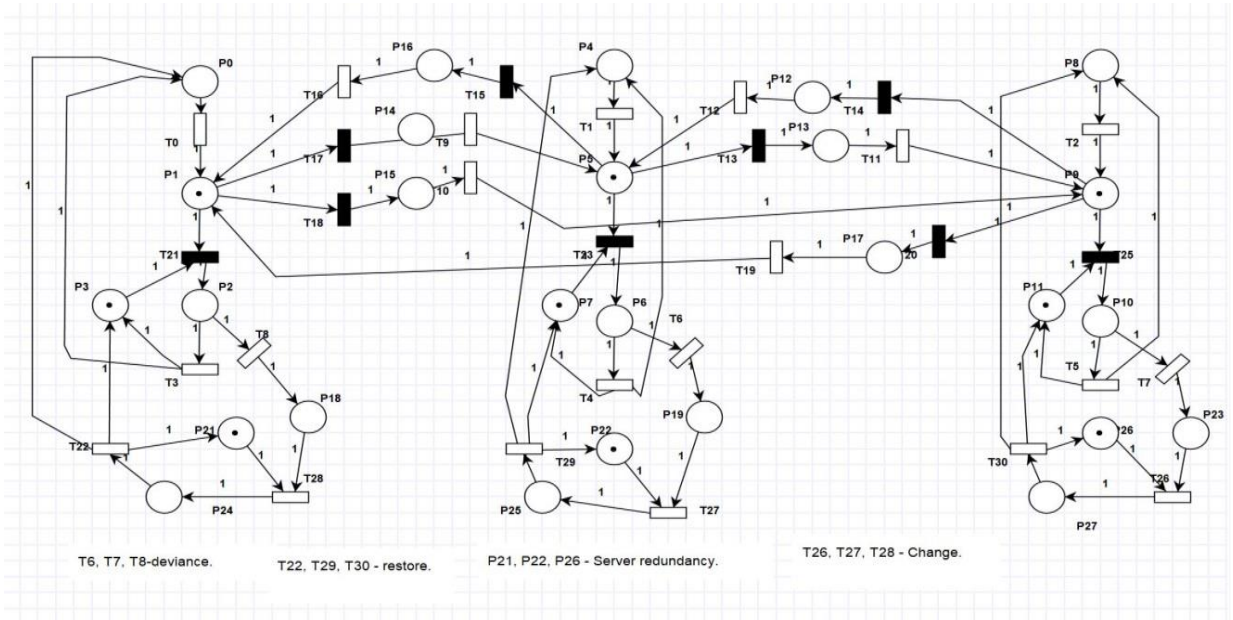


Figure 3.3. Hybrid topology modeling using Petri Nets

Description of the Hybrid topology scheme.

1. Also, as previously mentioned this topology is a mix of both the previous topologies, hence comes the name Hybrid; the structure of the proposed formation of controllers will contain six controllers.

2. There will be three main controllers working in parallel as one integrated entity just like the parallel topology hence, in this case parallel topology rules apply here.

3. Every node or part of the triple main controller will have its own backup controller which will be also connected to the network through two ways:

- Connected to its main controller to replace it in case of an attack on its main controller.

- Connected to the other backup controllers.

4. In case of an attack on any main controller it will be isolated alongside with the attacker's IP and it will be replaced with its substitute or backup controller till the maintenance of the infected controller finishes and of course before embarking that procedure; a bot will be sent to the attacking source. The Table 3.7 down below gives a description of the places of Petri Nets diagram.

Table 3.7. Description of Places

Place	Description
*P*1/*P*5/*P*9	Servers' allocation
*P*21/ *P*22/ *P*26	Server redundancy/backup servers
*P*2/ *P*6/ *P*10	Active processing
*P*18/ *P*19/ *P*23	Server/controller under attack
*P*24/ *P*25/ *P*27	Recovery of server/controller
*P*3/ *P*7/ *P*11	Processing next request
*P*0/ *P*4/ *P*8	Back to initial state
*P*12- *P*17	Sharing the information and updating the network configuration

The Table 3.8 contains a description of the transitions of the diagram of the Hybrid topology.

Table 3.8. Description of Transitions

Transition	Description
*T*0/ *T*1/*T*2	Transition from initial state to active processing
*T*21/ *T*23/ *T*25	Active processing
*T*3/ *T*4/ *T*5	Processing next request
*T*6/ *T*7/ *T*8	Deviance or attack state
*T*22/ *T*29/ *T*30	Transitioning to backup /restoring/ back to initial state
*T*26/ *T*27/ *T*28	Change
*T*9- *T*20	Sharing and updating the network configuration between servers

3.5.4 Ordinary Topology

As shown in the Figure 3.4, this modeling represents the usual ordinary topology with one controller and shows its weakness points.

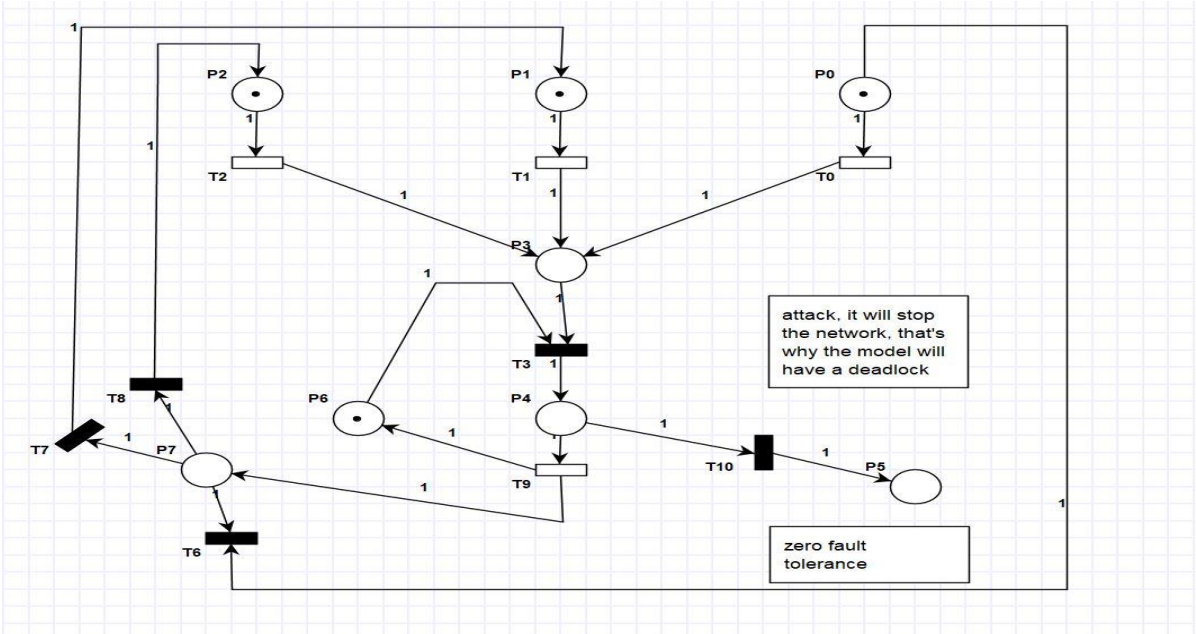


Figure 3.4. Ordinary topology modeling using Petri Nets

Description of the Ordinary topology scheme.

1. This topology is just representing the usual, simple, basic structure of Software-defined network using one controller.

2. It's just modeled for the sake of comparison to show how much effective our framework is with its proposed topologies.

3. This model shows how a one controller is really vulnerable and ineffective since there's a single point of failure SPOF which we want to overcome.

4. We have here one controller that processes switches' requests normally until an attack occurs.

5. In the case of an attack the above design shows that an attack can disrupt the controller and everything that relies on it since there's only one main controller. So, everything falls apart after infecting the controller and the whole network will be compromised. Which means that this topology has zero fault tolerance. The Table 3.9 shows the places of the diagram.

Table 3.9. Description of Places

Place	Description
*P*0/ *P*1/ *P*2	Selection of switches
*P*3	Main controller/ server
*P*4	Active processing
*P*6	Processing next request/ getting back to initial state
*P*7	Sending and receiving requests
*P*5	Attack on server/controller

The Table 3.10 describes the transitions of the Petri Nets diagram.

Table 3.10. Description of Transitions

Transition	Description
*T*0/ *T*1/ *T*2	Sending requests to controller
*T*3	Active processing
*T*9	Initial state/ replying to switches
*T*6/ *T*7/ *T*8	Selection of switches
*T*10	attack

3.6 Simulation of the proposed topologies using Generalized Stochastic Petri Nets module

Let's make a comparison between the four topologies described in sections 3.4.1 – 3.4.4 in terms of Average Number (distribution intensity) of Tokens on places that represent the SDN controllers. In this case, the tokens represent how many tasks or requests the controllers have to implement every 10 seconds, and the less tokens/requests in the controllers, would mean freer controllers hence, more reliable controllers; since it shows that the network controllers are less DoS/DDoS attacks prone and more capable of handling these attacks and dealing with them. In this comparison we left the weight ω of immediate transitions intact and gave the rate r of timed transitions a value of 0.1 since that we need those configurations of the network to be broadcasted every 10 seconds and that means that every 10 seconds the model state will change. The relationship between the time and weight of immediate transitions and rate of timed transitions, not to forget that weight is of a fixed value since it is immediate and that means that it will not wait to fire and the wait time is zero so, the focus will be on the rate value. That's why, if it's needed to make the time to be 10 seconds then, it's needed to change the rate value to be 0.1. The relationship would be:

$$\tau = \frac{1}{\omega} = \frac{1}{r}, \tag{3.19}$$

where τ represents the time, ω represents the weight of immediate transitions and r represents the rate of timed transitions.

This research, gave the models a fixed value of firings, meaning (the same fixed number of simulations attempts in the Generalized Stochastic Petri Nets GSPN module for each topology) for the transitions in each model which is 20 firings. Using the GSPN module in the PIPE software, it was obtained the results regarding the average tokens' number (distribution intensity or how many tokens exist in each place per unit of time) in the places that represent the SDN controllers. The results are presented in the Table 3.11 and the Figure 3.5.

Table 3.11. Average Number of Tokens (distribution intensity of tokens) in Places Representing SDN Controllers Using GSPN Module

No.	Places	Serial topology		Parallel topology		Hybrid topology		Ordinary topology	
		No. of controllers	Tokens Z_{Ki}	No. of controllers	Tokens Z_{Ki}	No. of controllers	Tokens Z_{Ki}	No. of controllers	Tokens Z_{Ki}
1	*P*1			1	0	1	0	1	2.0
2	*P*5			1	0	1	0		
3	*P*3	1	0.16						
4	*P*7	1	0.06						
5	*P*9			1	0	1	0		
6	*P*11	1	0.13						
7	*P*21					1	0.90		
8	*P*22					1	0.90		
9	*P*26					1	0.90		
Total		**3**	**0.36**	**3**	**0**	**6**	**2.71**	**1**	**2.0**

It should be known that the less average number of tokens (distribution intensity of tokens) within the PN places that represent the SDN controllers per unit of time, the less occupied PN places, which means that the SDN controllers will be less occupied and more capable to deter any DoS/DDoS attacks.

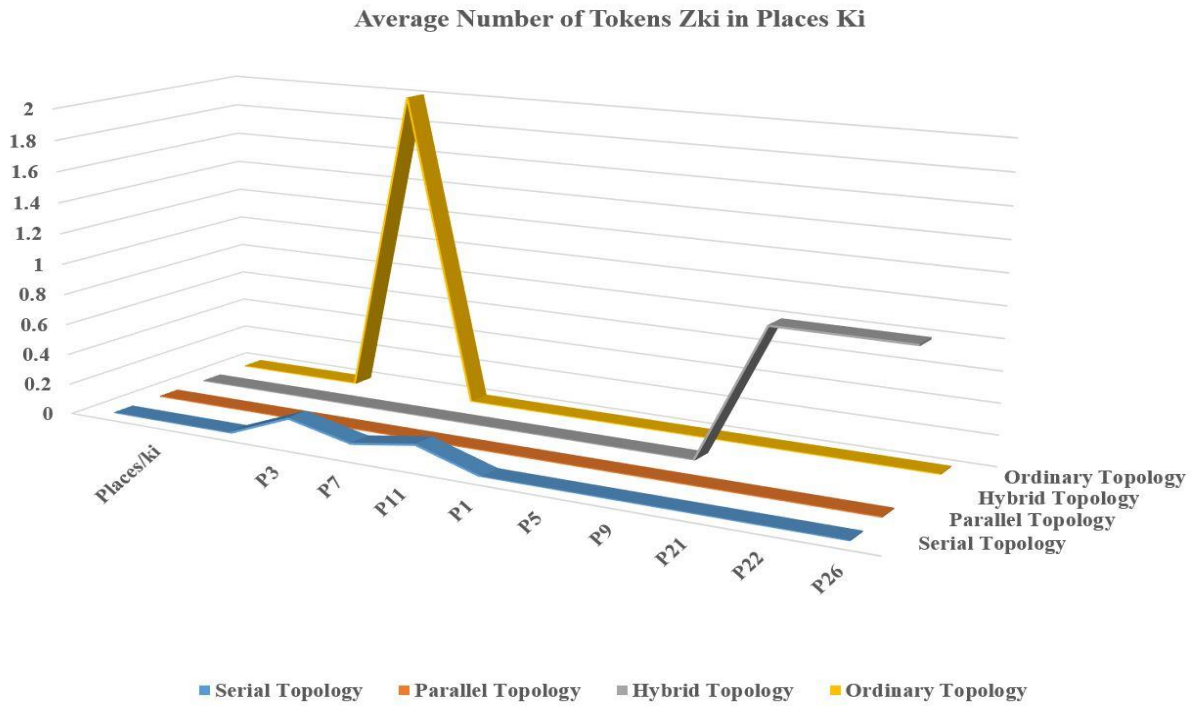


Figure 3.5. Comparison between topologies based on the average number (distribution intensity) of tokens in places representing the controllers in the Petri Nets model

From the information taken from the Table 3.11 and Figure 3.5, it is possible to infer that the least amount of tokens / requests lies within the controllers of the Parallel topology per unit of time, which means that they will be free and available for dealing with requests most of the time and that would mean that they will be more capable of deterring the DoS/DDoS attacks

hence, it is possible to say that the best topology of all the proposed topologies would be the Parallel topology while the worst, would be the already existing Ordinary Topology due to its high number or amount of tokens and to its dependence on a single controller that could be a point of failure.

3.7 Determining the efficiency of the proposed topologies using a set of parameters

The effect of the proposed topologies will be determined using the next parameters which are, Reliability of Service (RoS), Defense Factor (DF), Risk Factor (RF), Modified Risk Assessment (RM) and Cost effectiveness.

3.7.1 Computer networks Reliability of Service assuring using the proposed Topologies

The Table 3.12 and Figure 3.6 present the calculations and data regarding the values of TANR for different topologies that were calculated based on the Table 3.11. and by applying those numerical data in the RoS equation.

Table 3.12. The effect of multiple controllers' usage on *RoS* of the presented topologies

Topology	*TANR*	*RoS*
Serial	0.12	0.94
Parallel	0	1.0
Hybrid	0.45	0.78
Ordinary	2.0	

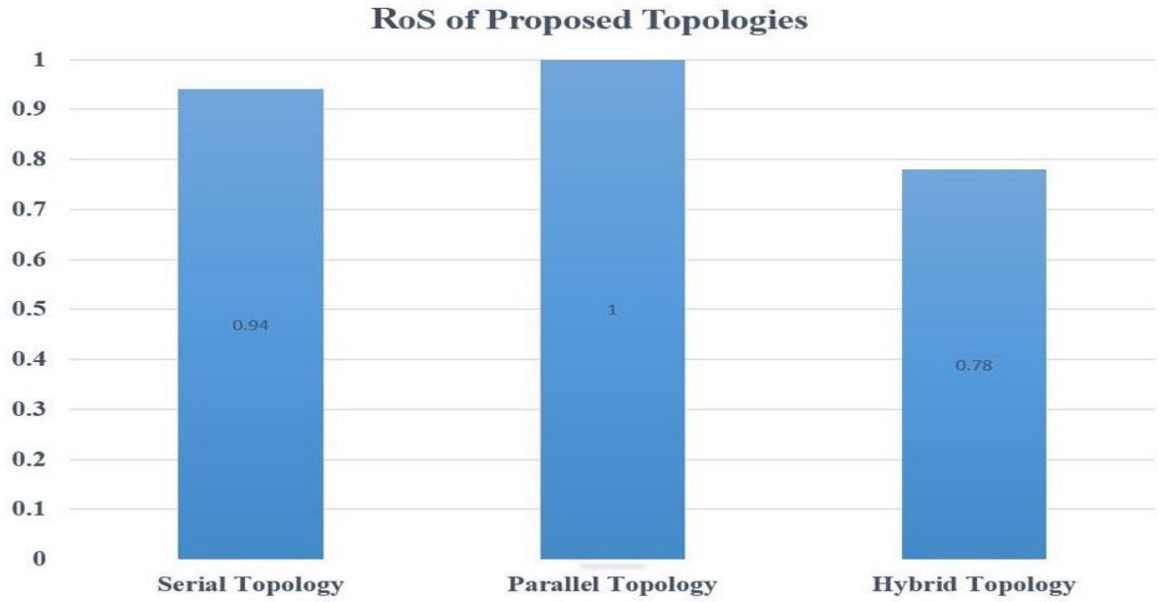


Figure 3.6. The values of RoS of different topologies

According to Table 3.12 and Figure 3.6, it is possible to conclude that, the Hybrid topology enhances the RoS by 78%, the Serial topology does better which is nearly 94% and the

Parallel topology enhances the RoS by 100%. We can notice that the best topology of all the described topologies will be the parallel topology since its model is empty of tokens most of time during which the firings took place and that means that this network topology was less occupied with tasks or its controllers were more available in unit of time, hence, more capable of providing a better data protection in terms of Reliability of Service (RoS).

3.7.2. Defense Factor formula for estimation of the security level of SDN topologies

After leveraging the Generalized Stochastic Petri Nets (GSPN) module in PIPE software to determine the data from table 3.11, we can use those acquired results to determine the relationship between the readings gained from the different simulations. Meaning the equation to assess the security level of networks especially the software-defined networks against cyberattacks especially those that could leverage the busy or flooded servers as a weakness point like DoS/DDoS attacks; we named this proposed equation as the network defense factor against cyberattacks law.

Before we use this law or equation, we need to describe the basis of this law itself and how and why it was formulated. First we have to emphasize that this law is formulated for different kinds of attacks but it is especially to measure risk assessment of DoS/DDoS attacks due to its main concept or feature that it depends on, which is how many operations are conducted by the controller/server, means how many requests that the controller is dealing with at a specific unit of time and as known and mentioned before that denial of service and distributed denial of service attacks DoS/DDoS are depending mainly on flooding the target with huge amounts of request packets to disrupt it or stop it completely so, the less the targeted device is occupied the better it is; because it will be more capable of dealing with that big amount of requests hence, it will have a better security defense level and longer time to respond to use its defense mechanisms like intrusion detection and intrusion prevention systems IDS/IPS, firewalls etc.

The law shows that the less requests a controller has the better security level and higher defense abilities it has and vice versa so, it’s an opposite relationship between the number of requests being handled at a specific unit of time and the Defense level assessment. And as known since this research focuses on assuring the security of the computer network generally and the security of the software-defined network in precise especially the control plane in its structure.

And there is the control plane represented by the controller as our main element of interest to secure and also the main component of the SDN that needs its security level to be

determined. Then it is of a great deal of importance to include that element mainly in the formula to figure out its security level. Hence, finding out the security level of the network itself. In other words, let defense factor DF be a function of *K* and *Z*:

$$DF=f\{K, Z\}, \tag{3.20}$$

where *K* is the number of controllers in the network represented by place in the PN model, and *Z* is the number of requests being served in each controller at the current unit of time, represented by the summation of all average numbers of tokens in all places that represent the topology controllers, meaning the distribution intensity of tokens and how many tokens are there together per unit of time in each controller; summed together for all controllers. If the relationship (3.9) is used with the Petri Net models to describe the relationship between them, then the result will be the equation (3.10), which represents the defense factor of a software-defined that can be estimated as:

$$DF= K / Z_i, \tag{3.21}$$

where *K* is the total number of the places that represent the controllers in a model $k= \{1\text{-}n\}$, and Z_K is the total value of existing tokens' intensity in these places, $Z_i = \{0 - \infty\}$. By applying the results from table 3.11 in the Defense Factor equation, we get the following results in Table 3.13 and Figure 3.7 that present a comparison between the gained Defense Factor results for different topologies.

Table 3.13. Comparison between Different SDN Topologies based on Their Defense Factor DF

Algorithm	**Serial Topology**	**Parallel Topology**	**Hybrid Topology**	**Ordinary Topology**
Defense factor *DF*	8.23	∞	2.21	0.50

So, from the numerical results of the Table 3.13, it is possible to conclude that all the proposed topologies have better defense ability against DoS/DDoS attacks than the single-controller Ordinary topology. The Serial topology has a DF value of 8.23, the Parallel topology proved to be the best of all the suggested topologies and it has nearly an optimal defense level against DoS/DDoS attacks and the Hybrid topology has a value of 2.21. The Figure 3.7 shows the Defense Factor values of different SDN topologies.

The numerical results of Risk Factor (*RF*) calculations which are presented in the Table 3.14 and Figure 3.8, and they are gained by applying the results from Table 3.11 in the Risk Factor (*RF*) equation.

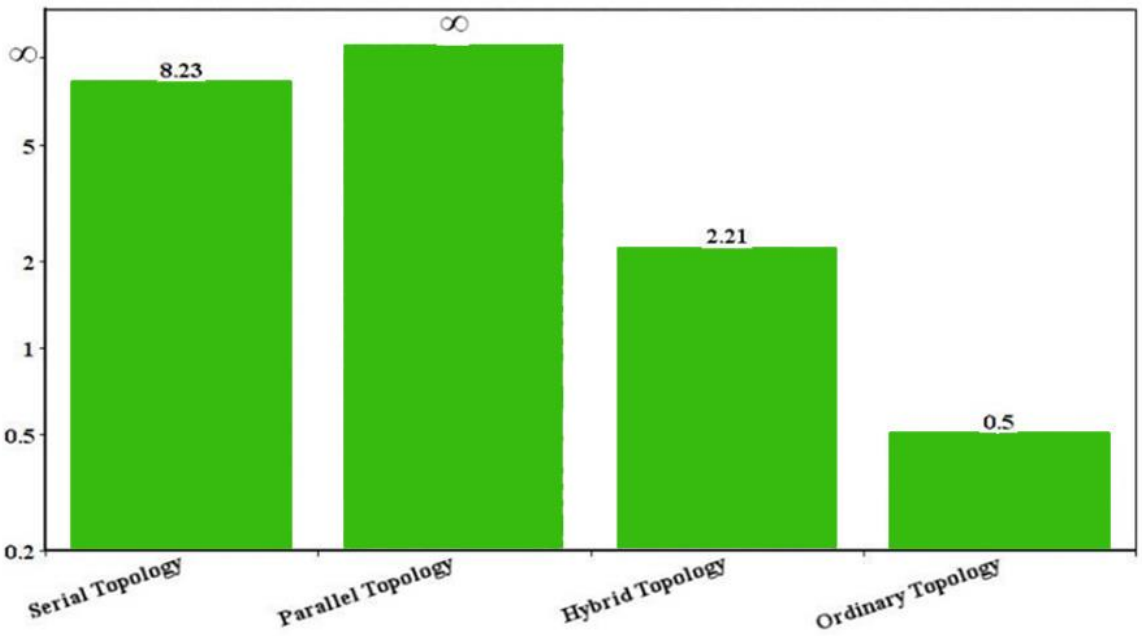


Figure 3.7. Defense Factor of different SDN topologies

Table 3.14. Comparison between Different SDN Topologies based on Their Risk Factor *RF*

Algorithm	Serial Topology	Parallel Topology	Hybrid Topology	Ordinary Topology
Risk factor *RF*	0.12	0	0.45	2

3.7.3. Risk Factor formula for estimation of the security level of SDN topologies

Based on the gained results from the simulation in PIPE software using GSPN module, a table was created showing the average amount or intensity of distribution of tokens in places representing the SDN controllers which in turn, represent the requests or operations dealt with by the SDN controllers, these recorded tokens were recorded only in the places that represent the SDN controllers as in Table 3.11. Based on this table, it was determined a relationship representing the behavior of the topologies in the simulation and this relationship was mapped mathematically to be as the defense factor formula.

Eventually, as seen, the defense factor law proves that the best topology is the Parallel topology from all the previously mentioned topologies; just like the result that was gained by the Petri Nets model.

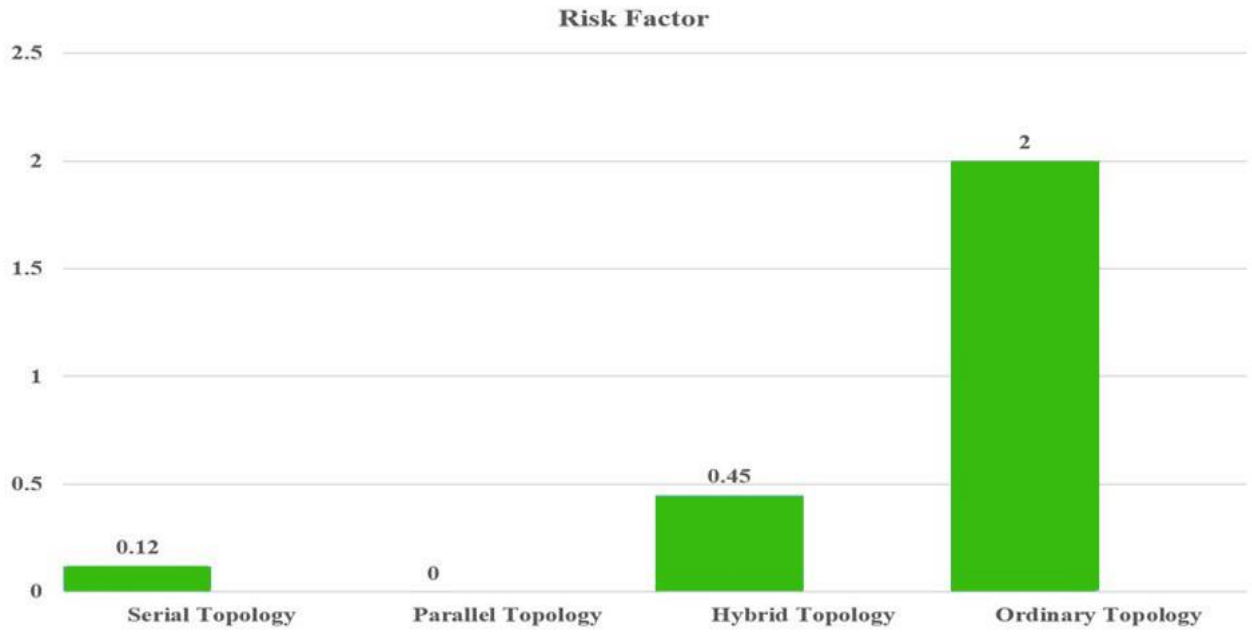


Figure 3.8. Risk Factor of different SDN topologies

Now of course the theoretical result may refer that the parallel topology has nearly the optimal performance and that it is an extrapolation in relation to the other topologies and based on the values gained from the simple modeling samples acquired by the PIPE Petri Nets software. In other words, that means that the experiments conducted refer to the parallel topology to have the least risk factor value hence, it will be considered as the most suitable one to be used by the SDN paradigm with the best ability to work under the pressure of cyber-attacks like DoS/DDoS attacks.

3.7.4. Modified Risk Assessment law of computer networks

By applying the results from Table 3.11 in the new Modified Risk Assessment (RM) equation, the Table 3.15 and Figure 3.9 will show the full analytical comparison of different SDN topologies based on the values of the new Modified Risk Assessment (RM).

Table 3.15. Comparison between different SDN topologies based on their Modified Risk Assessment values

No.	Topology	Modified Risk Assessment
1	Serial	0.3
2	Parallel	0
3	Hybrid	1.1
4	Ordinary	5.0

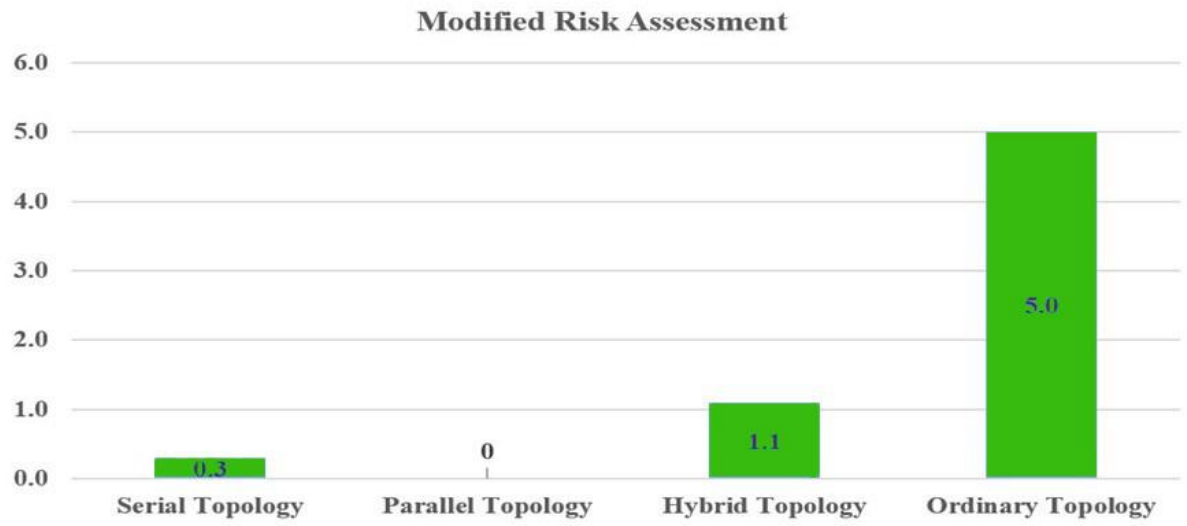


Figure 3.9. Modified Risk Assessment of different SDN topologies

As shown in Table 3.15 and Figure 3.9, it is possible to infer that the highest Risk Assessment value is for the single-controller Ordinary topology which means that it is the weakest of all topologies. While, the lowest probability value is for the Parallel topology which means that it is the best of all proposed topologies since it has the least value of vulnerability probability.

3.7.5. Evaluation of the Cost effect parameter (Y) of the SDN controllers' topologies

The evaluation of the Cost effect of the SDN security assurance was made taking into account that, *CC*= $2008.12, *SC*=$959, $1007.99, $1159 and *NC*=$ 649.99, $799.99, $1549.99. Based on this data, it was calculated the values of *Y1*, *Y2* and *Y3* in accordance with formulas (3.16 - 3.18). The results of those calculations are presented in Tables 3.16, 3.17, and 3.18 and in Figures 3.10, 3.11, and 3.12.

Table 3.16. The values of *Y1* at different *a* and *b*

a	***b***		
	1.29	2.51	3.08
1.73	1.27	1.39	1.63
1.99	1.28	1.40	1.44
2.09	1.28	1.41	1.45

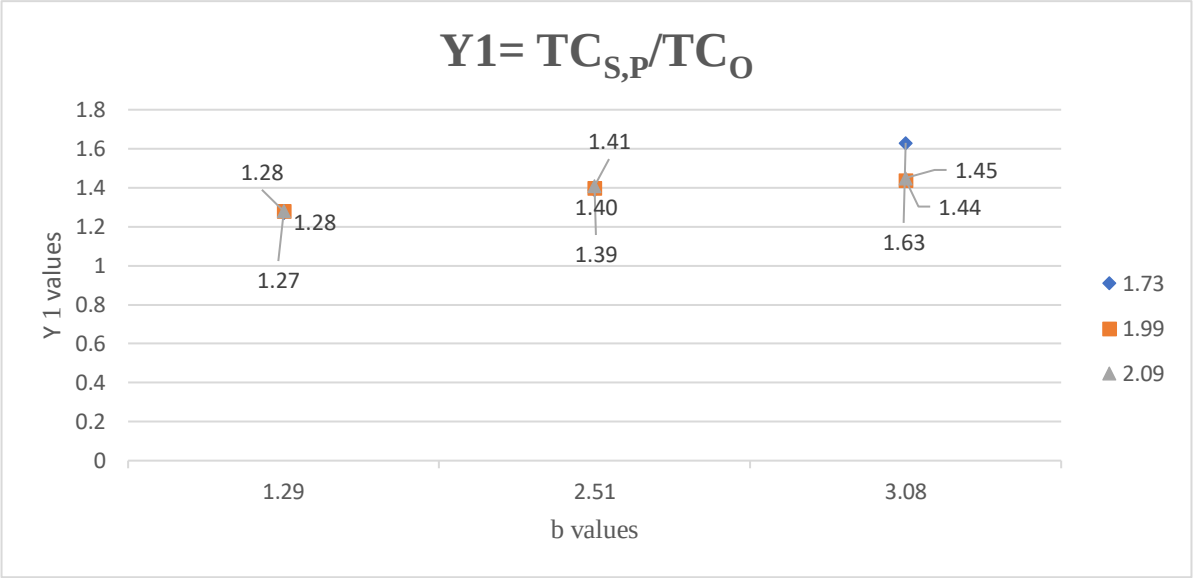


Figure 3.10. The values of Y1 at different a and b

Table 3.17. The values of *Y2* at different *a* and *b*

a	***b***		
	1.29	2.51	3.08
1.73	1.67	1.97	2.27
1.99	1.69	2.02	2.12
2.09	1.71	2.03	2.14

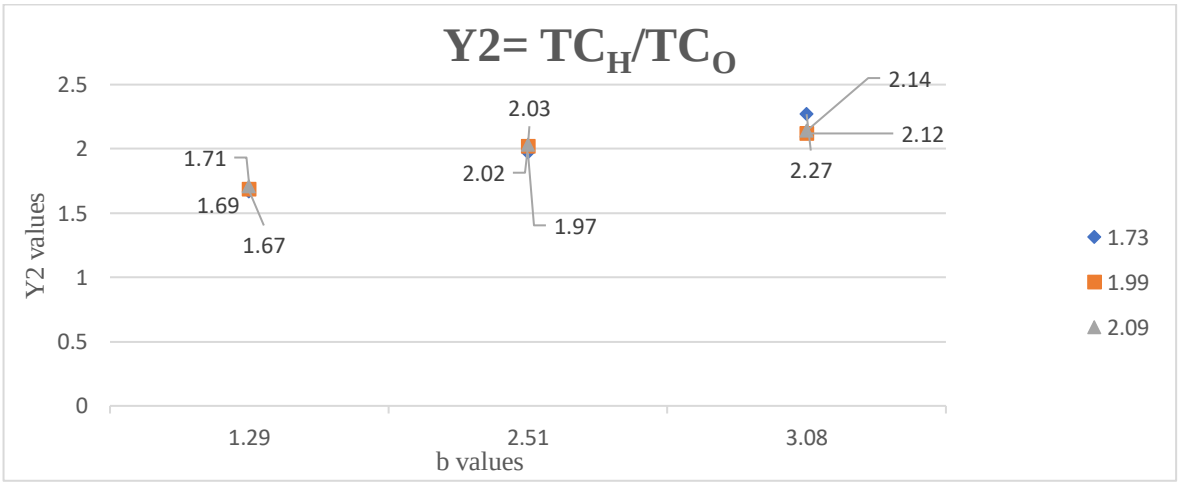


Figure 3.11. The values of Y2 at different a and b

Table 3.18. The values of *Y3* at different *a* and *b*

a	***b***		
	1.29	2.51	3.08
1.73	1.27	1.39	1.63
1.99	1.28	1.40	1.44
2.09	1.28	1.41	1.45

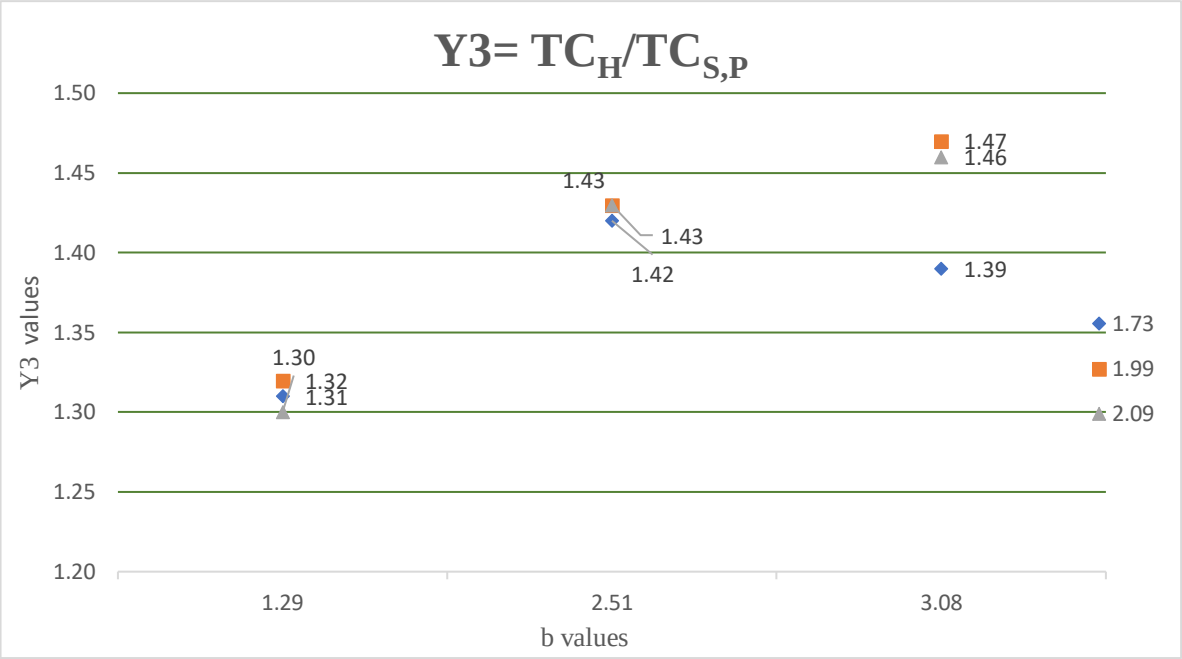


Figure 3.12. The values of Y3 at different a and b

The data presented in Figures 3.10 - 3.12 show that at increasing the values of *a = CC/SC* from 1.73 to 2.09 and at increasing the values of *b = CC/NC* from 1.29 to 3.08, the value of *Y1* is increasing from 1.27 to 1.63, value of *Y2* is increasing from 1.67 to 2.27 and value of *Y3* is increasing from 1.37 to 1.46.

It was inferred that the proposed serial and parallel topologies will increase the cost of security assurance by a maximum value of 1.63 times for selected prices of computer network equipment.

While, by using hybrid topology the cost of security assurance will increase to 2.27 times as compared to the cost of ordinary topology.

Also, the cost of security assurance using the hybrid topology, is higher by 1.46 times in comparison with serial and parallel topologies.

So, based on the Cost effect parameters, it is possible to conclude that it is better to use either the Serial or Parallel topologies.

3.7.6. Comparison of parameters of SDN topologies' security assessment

Eventually there is a need to provide an analytical comparison of the proposed parameters of computer networks security assessment based on the suggested topologies and the results of their simulation. The data are presented in Table 3.19 and Figure 3.13, where RoS is the Reliability of Service, DF is the Defense Factor, RF is the Risk Factor, RM is the Modified security Risk assessment and *Y* is the Cost effect parameter.

Table 3.19. The values of *RoS, DF, RF, RM* and Cost for the proposed topologies

No.	Topology	*RoS*	*DF*	*RF*	*RM*	*Y*
1	Serial Topology	0.94	8.23	0.12	0.3	Y1= 1.63
2	Parallel Topology	1.0	∞	0	0	Y1 = 1.63
3	Hybrid Topology	0.78	2.21	0.45	1.1	Y2 = 2.27
4	Ordinary Topology		0.50	2.0	5.0	

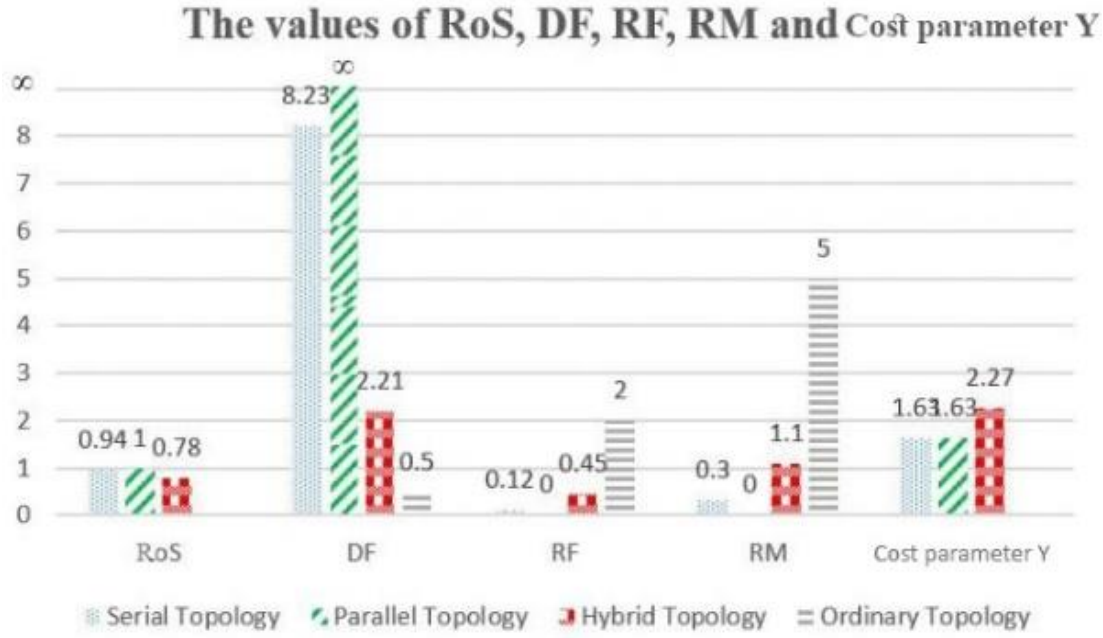


Figure 3.13. The values of RoS, DF, RF, RM and Cost for the proposed topologies

From the data presented in table 3.19 and figure 3.13, it is possible to infer that the Parallel topology is the best one from the point of view of all security estimation parameters, after that comes the Serial topology and the Hybrid topology is the last one.

The usage circumstances of these parameters differ from each other. When it is needed to measure the data protection performance then, it is possible to apply the RoS parameter, where it is possible to infer that the reliability of service of the serial topology is 0.94 of the previous Reliability of service of the single-controller ordinary topology, the parallel topology gained a 100% enhancement as compared to the ordinary topology and the hybrid topology proved to be better than the ordinary topology as well in terms of reliability of service by 0.78.

In other words, it is possible to say that the best topology of all the proposed topologies in terms of RoS, is the parallel topology.

When there's a need to determine the defense ability and the network's strength against DoS/DDoS attacks then, it is possible to depend on the DF parameter.

Where the defense factor that shows the strength of presented SDN topologies against DoS/DDoS attacks has the value of 8.23, while the parallel topology has a very high value close to ∞ since the number of the tokens that represent network requests in its controllers, are very small and close to 0 most of the processing time and that means that the defense factor value of this topology is close to optimal, meaning the parallel topology is nearly most reliable against DoS/DDoS attacks, the hybrid topology has a value of 2.21 and last but not least the ordinary topology has the least value which is 0.50, meaning that the single-controller topology has the weakest structure against DoS/DDoS attacks.

While if there was a need to reverse the situation and figure out the weakness of the network then, it is possible to reverse the parameter and use the RF which is the inversely proportional to DF. So that means the higher the value, the worse it is.

It is possible to conclude that the serial topology has the value of 0.12, the risk factor value of the parallel topology is 0 since it is nearly optimal and that means this structure has low risk, the hybrid topology has a risk factor of 0.45 and least but not last the ordinary topology has the highest value of 2.0, which refers to the high jeopardy that comes along the usage of this topology.

In case there was a need to measure the SDN environment weakness based on the computer networks law, then it is possible to modify it to be used based on the proposed SDN environments and their Petri Nets modeling and in this case the RM parameter can be used.

This parameter is a proof of how the DF equation has assured the security of the proposed topologies and reduced the risk value as compared to the already existing ordinary topology in terms of the risk assessment law that is used to describe the security risk for computer networks in general. So after modifying the risk assessment law by incorporating the effect of the Defense Factor in its equation to create a new parameter called the Modified Risk parameter, the research shows that the serial topology has a reduced RM value of 0.3, the parallel topology has a value of 0 as well hence, it is the best topology of all the suggested topologies, the hybrid topology's RM value is 1.1 and the single-controller topology has a value of 5.0 and this high RM value of the ordinary topology shows the security enhancements, the proposed topologies have in their design over the ordinary topology.

When there's a need to see the Cost effect of using the proposed topologies to assure the security of SDN paradigm as compared to the already-existing single-controller Ordinary topology; then it is possible to check the Cost parameter which shows that based on this research and on the specific prices' data gathered; that using of Serial and Parallel topologies prices could be increasing by maximum 1.63 times as compared to the usage of the Ordinary topology.

While using the Hybrid topology could increase the expenses by 2.27 times more than the expenses for the Ordinary topology.

To show a more detailed efficiency of the proposed SDN controllers' topologies, it was calculated the relations of the security assessment parameters of the new topologies as compared to the Ordinary topology as follows.

Relation of DF of Proposed topologies to the Ordinary topology $R_{DF} = DF_{pr}/DF_{O}$, Relation of RF of the Ordinary topology to the Proposed topologies $R_{RF} = RF_{O}/ RF_{pr}$, Relation of RM of the Ordinary topology to the Proposed topologies $R_{RM} = RM_{O}/RM_{pr}$. The results of the calculations are presented in Table 3.20.

Table 3.20. The values of parameters for the proposed topologies in comparison with the Ordinary topology

No.	Topology	*RoS*	R_{DF}	R_{RF}	R_{RM}	*Y*
1	Serial Topology	0.94	16.46	16.66	16.66	1.63
2	Parallel Topology	1.0	∞	∞	∞	1.63
3	Hybrid Topology	0.78	4.42	4.44	4.54	2.27

From the Table 3.20, it is possible to infer that Serial topology has a better RoS by 0.94, better R_{DF} by16.46 times, less R_{RF} and R_{RM} by 16.66 times and more cost *Y* by 1.63 as compared to the Ordinary topology.

On the other hand, the Parallel topology has a better RoS by 100%, better R_{DF} by ∞ times, less R_{RF} and R_{RM} by ∞ times and more cost *Y* by 1.63 as compared to the Ordinary topology. While the Hybrid topology has a better RoS by 0.78, better R_{DF} by 4.42 times, less R_{RF} by 4.44 times, less R_{RM} by 4.54 times and more cost *Y* by 2.27 as compared to the Ordinary topology.

The data presented in Table 3.20, show that the Parallel topology can be characterized as the topology with the highest value of security protection in comparison with the other topologies.

The Serial topology will assure the security level by more than 16 times. Regarding the cost of the proposed topologies, it was concluded that the Serial and Parallel topologies require the increasing of the cost by 1.63 times as compared to the Ordinary topology and the Hybrid topology by 2.27 times as compared to the Ordinary topology.

The novelty of these proposed parameters stems from their basis, which is the modeling of the suggested SDN topologies using Petri Nets (PN) system to simulate their behavior and to

acquire numerical results from that. Based on these numerical results, those parameters were provided.

Alongside the algorithms, topologies, and their Petri Nets modeling, this research provides mathematical mechanisms to evaluate the security level with the data protection performance of computer networks based on SDN technologies.

3.8 Conclusions of Chapter 3

1. It was made the analysis of the existing approaches for evaluation of computer networks security level, which shows that to the best of our knowledge there aren't researches that permit the effective evaluation of the security level assurance of the elaborated SDN controllers' topologies.

2. It is elaborated and described a new method for evaluation of computer networks security level, based on Petri Nets applications and a set of parameters that were proposed in this research and they are, Reliability of Service (RoS), Defense Factor (DF), Risk Factor (RF), Modified Risk assessment parameter (RM) and the Cost effect of the Proposed SDN controllers' topologies in case of using them.

3. It was made the modeling of the proposed Serial, Parallel, and Hybrid SDN topologies using Petri Nets and they were compared to the already existing single-controller Ordinary topology.

4. It was made the simulation of the proposed topologies using Generalized Stochastic Petri Nets (GSPN). It was shown that it is possible to infer that the least amount of tokens / requests lies within the controllers of the Parallel topology per unit of time, which means that they will be free and available for dealing with requests most of the time and that would mean that they will be more capable of deterring the DoS/DDoS attacks hence, it is possible to say that the best topology of all the proposed topologies would be the Parallel topology while the worst, would be the already existing Ordinary Topology due to its high number or amount of tokens and to its dependence on a single controller that could be a point of failure.

5. It was made the evaluation of the efficiency of the proposed topologies using a set of the proposed parameters which are, Reliability of Service (RoS), Defense Factor (DF), Risk Factor (RF), Modified Risk Assessment (RM) and Cost Effect.

6. It was established that the RoS parameter of the Hybrid topology is 78%, of the Serial topology is nearly 94%, and of the Parallel topology enhances the RoS is of 100%. It is possible to notice that the best topology of all the described topologies will be the Parallel topology since its model is empty of tokens most of time during which the firings took place and that means that

this network topology was less occupied with tasks or its controllers were more available in unit of time, hence, more capable of providing a better data protection in terms of Reliability of Service (RoS).

7. It was established that the best DF value is for the Parallel topology as compared to the other proposed topologies and as compared to the existing Ordinary topology.

8. It was determined that it is possible to infer that the highest Risk Factor value is for the single-controller Ordinary topology which means that it is the weakest of all topologies. While, the lowest probability value is for the Parallel topology which means that it is the best of all proposed topologies since it has the least value of vulnerability probability.

9. Regarding the Modified Risk parameter (RM), the research shows that the serial topology has a reduced RM value of 0.3, the parallel topology has a value of 0 as well hence, it is the best topology of all the suggested topologies, the hybrid topology's RM value is 1.1 and the single-controller topology has a value of 5.0 and this high RM value of the ordinary topology shows the security enhancements, the proposed topologies have in their design over the ordinary topology.

10. It was concluded that the Serial and Parallel topologies would have more cost in establishing as compared to the Ordinary topology by 1.63 times and the Hybrid topology's cost increases by 2.27 times as compared to the Ordinary topology.

11. The novelty of these proposed parameters stems from their basis, which is the modeling of the suggested SDN topologies using Petri Nets (PN) system to simulate their behavior and to acquire numerical results from that. Based on these numerical results, those parameters were provided.

12. Alongside the algorithms, topologies, and their Petri Nets modeling, this research provides mathematical mechanisms to evaluate the security level with the data protection performance of computer networks based on SDN technologies.

13. The usage circumstances of these parameters differ from each other. When it is needed to measure the **data protection performance** then, it is possible to apply the **RoS** parameter. When there's a need to determine the **defense ability** and the network's strength against DoS/DDoS attacks then, it is possible to depend on the **DF parameter**. While if there was a need to determine the **weakness of the network** then, it is possible to use the parameter **RF.** In case there was a need **to measure the SDN environment weakness** based on the computer networks law, in this case the **RM** parameter can be used. Also, if there was a need to determine the **cost of the security assurance of the network**, then it is possible to use the proposed **Cost parameter** and its formulas.

14. The gained mathematical results are mostly theoretical and it is well considered that division over zero is not permissible but due to the results gained by the PIPE software simulation; it was imperative to conduct such a mathematical operation, but this shows that the Parallel topology is very reliable and near optimal.

GENERAL CONCLUSIONS AND RECOMMENDATIONS

GENERAL CONCLUSIONS

Important scientific solved problem is elaboration of a new suite of algorithms and SDN controllers' topologies to increase the security level of SDN and elaboration of the theoretical assessment of computer networks' security level.

The main scientific results obtained in the research are the following

1. In the thesis, it was argued the Software-Defined Networking as a potential solution for addressing cyber threats. It was presented a security efficiency evaluation of some of the most prominent SDN-related techniques, researches and methods, which shows the importance of SDN in the technology and its ability to be incorporated with other technologies to enhance them and the flexibility of SDN to accept other technologies in its paradigm to gain more enhancements [21].

This analysis permitted to formulate the problems noticed in the SDN environmental structure, such as Centralization - despite that SDN controllers give the ability to manage the network environment from a single point but that also could be considered as a weakness point if jeopardized and used as a single point of failure.

2. The new cryptographic algorithms were proposed, integrated within technologies for assuring SDN security. The algorithms suite consists of the Hydra framework mainly and integrated within it, the secured channel of VPN algorithm, Double RSA algorithm, and Distributed ledger of Blockchain algorithm. These algorithms work together and interact to form the Hydra-like behavior of the framework to deter Man In The Middle (MITM) attacks and Denial of Service/Distributed Denial of Service (DoS/DDoS) attacks [22].

3. Three topologies of SDN controllers were proposed, which are serial, parallel and hybrid ones, for increasing the security level of networks by adding more controllers that interact in a specific way, to address the issue of Single Point Of Failure (SPOF) [23], [24], [25].

4. The new method proposed for evaluation of the computer networks security level, based on using of the Petri Nets and a set of parameters such as Reliability of Service (RoS), Defense Factor (DF), Risk Factor (RF), Modified Risk assessment parameter (RM), and Cost Effect (Y) [26].

5. It was established, that when it is needed to measure the data protection performance then, it is possible to apply the RoS parameter. When there's a need to determine the defense ability and the network's strength against DoS/DDoS attacks then, it is possible to depend on the DF parameter. While if there was a need to determine the weakness of the network then, it is

possible to use the parameter RF. In case there was a need to measure the SDN environment weakness based on the computer networks law, in this case the RM parameter can be used [27].

6. The modeling and simulation of the proposed Serial, Parallel, and Hybrid SDN topologies using Petri Nets and Generalized Stochastic Petri Nets (GSPN), was made. It was shown that it is possible to infer that the least number of tokens/requests lies within the controllers of the Parallel topology per unit of time, which means that they will be free and available for dealing with requests most of the time and that would mean that they will be more capable of deterring the DoS/DDoS attacks hence, it is possible to say that the best topology of all the proposed topologies would be the Parallel topology [28], [29].

7. It was determined, that the Serial and Parallel topologies would have more cost in establishing as compared to the Ordinary topology by 1.63 times and the Hybrid topology's cost increases by 2.27 times as compared to the Ordinary topology [30].

The main research result is assuring the security of Software-Defined Networks by solidifying its structure against cyber-attacks especially; Denial of Service / Distributed Denial of Service (DoS/DDoS) and Man In The Middle (MITM) attacks; by securing the Single Point Of Failure (SPOF) through the usage of multi-controller topologies that interact with each other in a specific way and by using the Hydra framework to secure the connection between those multiple controllers in the proposed topologies; respectively.

The second obtained result is that the proposed topologies are more deterrent and more capable of defending themselves against DoS/DDoS attacks as compared to the ordinary single-controller topology and more capable of recovering themselves since they will be emptier per unit of time than the controller of the ordinary topology hence; less prone to fake requests flooding of DoS/DDoS attacks. Among those proposed topologies the best of them is the parallel topology since its controllers are emptier than those of the other two topologies.

The third obtained result is gained by having a mathematical equation that could be used as a measurement tool for the security risk level of computer networks that leverage the SDN paradigm especially; those that are based on any of the proposed topologies in this research. That means that it could be used as a methodological mathematical tested for network components manufacturers and developers for revealing the feasibility of the future software-based and hardware-based products before releasing them into the technical market.

The novelty of the research consists of elaborated new algorithms for assuring the security of SDN, which are Hydra, Double RSA, and distributed ledger of Blockchain; the elaboration of new controllers' topologies, which are Serial, Parallel and Hybrid topologies; the elaboration of a new method of security evaluation of computer networks based on Petri Nets

and five parameters which are Reliability of Service, Defense Factor, Risk Factor, the Modified Risk assessment law and the Cost effect of the proposed SDN controllers' topologies.

Applicative value of the work is determined by the developed framework, which has a big contribution for the SDN community by proposing new SDN topologies to deal with the centralization issue and by protecting the connection between multiple SDN controllers. Also, provides a better view for the security level of a specific network by measuring it using various mathematical tools that are based on different proposed parameters.

RECOMMENDATIONS.

The present research can be expanded further more; since that developing a framework could make use of the support of the right practical tools and network components; whether software-based or hardware-based. The research could be stretched and investigated deeper in various directions, like:

1. The possibility to enhance the usage circumstance for the proposed parameters and make them more often used and more capable to measure the security level of the data plane of the software-defined network structure.
2. Securing TLS/SSL against DoS, as know the controller communicates with the switches through the transport layer using protocols like openflow protocol and that kind of connection is considered as the southbound protocol since that controllers connect with each other using the east-westbound API and connect with the apps of the management plane using the northbound protocols so; since that in this research it was assured the security of the east-westbound API; it's possible to investigate the security of the southbound protocols to secure the openflow protocol by securing the transport layer security (TLS) and its successor; secure socket layer (SSL).
3. Using neural networks and artificial intelligence to create an analytical algorithm for detecting anomalies based on the statistics, and that algorithm could be built on top of a framework that is integrated with the controller.
4. Usage of data mining to calculate mistake, problems, attacks, network issues, code errors and human mistake; all those statistics combined with the previous article of AI to create a self-sufficient and self-protecting computer network that uses the SDN structure.
5. Adding more controllers in the proposed topologies could considered as enhancement, especially if there were more new simulations using different approaches to gain new better results.

BIBLIOGRAPHY

1. OUDIN, R. Simply SDN, OFLOPS-SUME and the art of switch characterization, In: *IEEE Journal on Selected Areas in Communications*, 2018, 1(99), pp. 1-1.
2. KARNOUSKOS, S. Stuxnet Worm Impact on Industrial Cyber-Physical System Security. In: *IECON 2011-37th Annual Conference of the IEEE Industrial Electronics Society*, 2011, pp. 4490-4494.
3. Hacker collective. [Online]. 2019, [Cited: 10.05.2019] available: https://en.wikipedia.org/wiki/L0pht.
4. Internet blocking. [Online]. 2019, [Cited: 12.05.2019] available: https://netblocks.org/reports/study-shows-extent-of-iraq-internet-shutdown-and-social-media-restrictions-during-protests-zPyXjzAE.
5. OSAGIE, M, ENAGBONMA, O, INYANG, A. The Historical Perspective of Botnet Tools, In: *Current Journal of Applied Science and Technology*, 2019, 32(6), pp. 1-8.
6. PRASAD, K, REDDY, A, RAO, K. DoS and DDoS Attacks: Defense, Detection and Trace Back Mechanisms – A Survey. In: *Global journal of computer science and technology: E Network, Web and Security*, 2014, 14 (7), pp. 15-32.
7. MALIK, A, AHSAN, A, SHAHADAT, M, TSOU, J. Man-in-the-middle-attack: Understanding in simple words. In: *International Journal of Data and Network Science*, 2019, 3, pp. 77-92.
8. BULLEE, J, MONTOYA, L, et. al. Spear Phishing in Organizations explained. In: *Information & Computer Security*, 2017, 25 (1), pp. 593-613.
9. ALDWAIRI, M, HASAN, M, BALBAHAITH, Z. Detection of Drive-By Download Attacks Using Machine Learning Approach. In: *International Journal of Information Security and Privacy*, 2017, 11(4), pp. 1-14.
10. CHESTER, J. Analysis of Password Cracking Methods & Applications. In: *Honor Research Projects*, 2015, 7, pp. 1-15.
11. AHMAD, K. Classification of SQL Injection attacks. In: *VSRD Technical & none-Technical Journal*, 2010, 1(4), pp. 235-242.
12. GUPTA, S, GUPTA, B. Cross-Site Scripting (XSS) attacks and defense mechanisms: classification and state-of-the-art. In: *International Journal of systems assurance Engineering and management*, 2015, 8, pp. 512-530.
13. SUBRAMANIYASWAMY, V, KALYANI, G, LIKHITHA, N. Securing Web Applications from Malware Attacks Using Hybrid Feature Extraction. In: *International Journal of Pure and Applied Mathematics*, 2018, 119 (12), pp. 13367-13385.

14. DAI, H, WANG, Q, et. al. On Eavesdropping Attacks in Wireless Sensor Networks with Directional Antennas. In: *International Journal of distributed Sensor Networks*, 2013, 9(8), pp. 1-13.
15. PATE, K, MAHIDA, D. Birthday Attack on Cryptography Applications Concept and Real Time uses. In: *International Journal of Advance Engineering and Research Development*, 2017, 4(72). pp. 1-5.
16. NAMANYA, A, CULLEN, A, AWAN, I, PAGNA DISS, J. The World of Malware: An Overview. In: *IEEE International Conference on Future Internet of Things and Cloud*, 2018, University of Bradford, 6, pp. 420-427.
17. WHEATLEY, S, MAILLART, T, SORNETTE, D. The Extreme Risk of Personal Data Breaches and the Erosion of Privacy. In: *The European Physical Journal B*, 2016, 89 (7), pp. 1-12.
18. NESHENKO, N, BOU-HARB, E, CRICHIGNO, J, et. al. Demystifying IoT security: An Exhaustive Survey on IoT Vulnerabilities and A First Empirical Look on Internet-scale IoT Exploitations. In: *IEEE Communications Surveys & Tutorials*, 2019, 21(3), pp. 2702-2733.
19. BISWAS, A, GHOSAL, D, NAGARAJA, S. A Survey of Timing Channels and Countermeasures. In: *ACM Computing Surveys*, 2017, 50(1), pp. 1-39.
20. JYOTHI, K, Dr. REDDY, B. Study on Virtual Private Network (VPN), VPN's Protocols and Security. In: *International Journal of Scientific Research in Computer Science*, 2018, 3(5), pp. 919-932.
21. AZIZ, D. The Importance of VLANs and Trunk links in Network Communication Areas. In: *International Journal of Scientific and Engineering Research*, 2018, 9(9), pp. 10-15.
22. TRABELSI, Z, ZEIDAN, S, et. al. Statistical Dynamic Splay Tree Filters towards Multilevel Firewall Packet Filtering Enhancement. In: *Computers & Security*, 2015, 53, pp. 109-131.
23. KENNEDY, E, EUKU, M, SSEKIBUULE, R. Packet Drop Attack Detection Techniques In Wireless Ad Hoc Networks: A Review. In: *International Journal of Network Security & its applications*, 2014, 6(5), pp. 75-86.
24. MAHENDIRAN, A, APPUSAMY, R, KARTHIK, S. Intrusion Detection and Prevention System: Technologies and Challenges. In: *International Journal of Applied Engineering Research*, 2015, 10(87), pp. 1-12.

25. ANTONELLO, R, FERNANDES, S, et. al. Deep Packet Inspection Tools and Techniques In Commodity Platforms: Challenges and Trends. In: *Journal of Network and Computer Applications*, 2012, 35(6), pp. 1863–1878.
26. WAIRISAL, M, SURANTHA, N. Design and Evaluation of Efficient Bandwidth Management for A Corporate Network. In: *International Conference on Information Management and Technology*, 2018, Bina Nusantara University, pp. 98-102.
27. THELIS, R, et. al. Access Agent: Improving the Performance of Access Control Lists. In: *International of Scientific & Technology Research*, 2016, 5(4), pp. 143-150.
28. SANKARDAYAL, P. An Impression of Software Defined Networks and OpenFlow for Beginners, In: *IJREAT International Journal of Research in Engineering & Advanced Technology*, 2014, P.S.R. Engineering College, 1(6), pp. 1-4.
29. PAVEL, P, et. al. Modern security issues in Software-Defined Networking. In: *Information technologies, systems and networks*, 2017, Lomonosov Moscow State University, 1(5), pp. 182-187.
30. IP knowledge. (2013). traditional VS SDN [whitepaper].
31. SCOTT-HAYWARD, S, O'CALLAGHAN, G, SEZER, S. SDN Security: A Survey. In *IEEE SDN for Future Networks and Services (SDN4FNS)*, 2013, Centre for Secure Information Technology (CSIT), (10), pp. 1-7.
32. DABBAGH, M, HAMDAOUI, B, GUIZANIY, M, RAYES, A. Software-Defined Networking Security: Pros and Cons. In: *IEEE Communications Magazine*, 2015, Oregon State University, 53 (6), pp.73 – 79.
33. Software-defined networking (SDN) market size worldwide from 2013 to 2021. [Online]. [Cited: 11.09.2018]. Available: https://www.statista.com/statistics/468636/global-sdn-market-size/.
34. QIANG, H, SHENGBAO, W. A Low-Cost Measurement Framework in Software Defined Networks. In: *Journal of information security*, 2017, 10, pp. 54-66.
35. VAN ADRICHEM, N, DOERR, C, KUIPERS, F. OpenNetMon: Network Monitoring in Openflow Software Defined Networks. In: Proc. *IEEE Network Operations and Management Symposium (NOMS)*. 2014, pp. 1-8.
36. CHOWDHURY, S, BARI, M, AHMED, R, BOUTABA, R. Payless: A Low-Cost Network Monitoring Framework for Software Defined Networks. In: Proc. *14 IEEE/IFIP Network Operations and Management Symposium (NOMS)*, 2014, pp.1-9.

37. YU, C, LUMEZANU, C, ZHANG, Y, SINGH, V, JIANG, G, MADHYASTHA, H. FlowSense: Monitoring Network Utilization with Zero Measurement Cost. In: *Passive and Active Measurement*, Springer, 2013, pp. 31-44.
38. SUH, J, KWON, T, DIXON, C, FELTER, W, CARTER, J. OpenSample: A Low-Latency Sampling-Based Measurement Platform for Commodity SDN. In: *International Conference on Distributed Computing Systems*, 2014, pp. 228-237.
39. YU, M, JOSE, L, MIAO, R. Software Defined Traffic Measurement with OpenSketch. In: Proc. *10th USENIX Symposium on Networked Systems Design and Implementation*, 2013, 13, pp. 29-42.
40. KOGOS, K, SOKOLOV, A. Methods of IPD normalization to counteract IP timing Covert channels. In: *International conference "information technology and nanotechnology 2017"*, 2017, Moscow Engineering Physics institute, (3), pp. 118-126.
41. ANYI, Liu, CHEN, J, Wechsler, H. Real-Time Timing Channel Detection in a Software-Defined Networking Virtual Environment. In: *Intelligent Information Management*, 2015, 7 (6), pp. 283-302.
42. What are the 12 biggest cloud computing security threats? [Online] [Cited: 11.10.2018]. Available: https://www.ibm.com/blogs/cloud-computing/2016/04/01/12-biggest-cloud-computing-security-threats/.
43. BERK, V., GIANI, A, CYBENKO, G. (2005) Covert Channel Detection Using Process Query Systems. In: Proc. *FLOCON-CERT*, Pittsburgh, 20-22 September 2005.
44. CABUK, S. IP Covert Timing Channels: Design and Detection. In: Proc. *The 11th ACM Conference on Computer and Communications Security*, Washington DC, 2004, pp. 178-187.
45. SHAH, G, MOLINA, A, BLAZE, M. Keyboards and Covert Channels. In: Proc. *The 15th USENIX Security Symposium*, Vancouver, 2006, pp. 59-75.
46. GIFFIN, J, GREENSTADT, R, LITWACK, P, TIBBETTS, R. Covert Messaging through TCP Timestamps. In: Proc. *The 2nd International Conference on Privacy Enhancing Technologies*, San Francisco, 2002, pp. 194-208.
47. ALSHNTA, M, ABDOLLAH, M, AL-HAIQI, A. SDN in the home: A survey of home network solutions using Software Defined Networking. In: *Journal of Cryptologia*, 2018, 5(1), pp. 1-40.
48. User centered home networking. [Online] [Cited: 20.11.2017]. available: http://homenetworks.ac.uk/

49. MORTIER, R, BEDWELL, B, GLOVER, K, et. al. Supporting novel home network management interfaces with openflow and NOX. In: Proc. *The ACM SIGCOMM 2011 Conference*, 2011, 41, pp. 464–465.
50. MORTIER, R, RODDEN, T, LODGE, T, MCAULEY, D, et. al. Control and understanding: Owning your home network. In: Proc. *Fourth International conference on communication systems and networks (COMSNETS 2012)*, 2012, pp. 1–10.
51. CHETTY, M, & FEAMSTER, N. Refactoring network infrastructure to improve manageability: A case study of home networking. In: *Computer Communication Review (CCR)*, 2012, 42(3), pp. 54–61.
52. BOUSSARD, M, BUI, D, DOUVILLE, R, et. al. The majord'home: A SDN approach to let ISPs manage and extend their customers' home networks. In: *10th International conference on network and service management (CNSM) and workshop*, 2014, pp. 430–433.
53. Moyano, R, Fernández, D, Bellido, L, González, C. A software-defined networking approach to improve service provision in residential networks. In: *International Journal Network Management*, 2017, 27(6), pp. 1-19.
54. ABUTEIR, R, FLADENMULLER, A, FOURMAUX, O. An SDN approach to adaptive video streaming in wireless home networks. In: *International wireless communications and mobile computing conference (IWCMC)*, 2016, pp. 321–326.
55. YANG, H, WANG, X, NGUYEN, C, LU, S. Conan: Content-aware access network flow scheduling to improve QoE of home users. In: Proc. of *The 2016 ACM SIGCOMM conference*, 2016, pp. 575–576.
56. JANG, H, HUANG, C, YEH, F. Design a bandwidth allocation framework for SDN based smart home. In: *IEEE 7th annual information technology*, electronics and mobile communication conference (IEMCON), 2016, pp. 1–6.
57. EGHBALI, H, WONG, V. Bandwidth allocation and pricing for SDN-enabled home networks. In: *IEEE international conference on communications (ICC)*, 2015, pp. 5342–5347.
58. HERNANDO, A, FARIÑA, A, et. al. Virtualization of residential IoT functionality by using NFV and SDN. In: *IEEE international conference on consumer electronics (ICCE)*, 2017, pp. 86–87.
59. GHARAKHEILI, H, EXTON, L, SIVARAMAN, V, et. al. Third-party customization of residential Internet sharing using SDN. In: *International telecommunication networks and applications conference (ITNAC)*, 2015, pp. 214–219.

60. KIM, H, SUNDARESAN, S, CHETTY, M, et. al. Communicating with caps: Managing usage caps in home networks. In: Proc. *The ACM SIGCOMM conference*, 2011, pp. 470–471.
61. WANI, A, REVATHI, S. Analyzing Threats of IoT Networks Using SDN Based Intrusion Detection System (SDIoT-IDS), In: *Journal of Cryptology*, 2018, pp. 536-542.
62. THANIGAIVELAN, N, NIGUSSIE, E, KANTH, R, et. al. Distributed internal anomaly detection system for internet-of-things. In: *13th IEEE Annual Consumer Communications Networking Conference (CCNC)*, 2016, pp. 319–320.
63. PONGLE, P, CHAVAN, G. Real time intrusion and wormhole attack detection in Internet of Things. In: *International Journal of Computer Applications*, 2015, 121(9), pp.1–9.
64. AL-SHAIKHLI, A, CEKEN, C, AL-HUBAISHI, M. WSANFlow: An Interface Protocol Between SDN Controller and End Devices for SDN-Oriented WSAN. In: *Journal of Cryptology*, 2018, 101, pp. 755-773.
65. LUO, T, TAN, H, QUEK, T. Sensor openflow: Enabling software-defined wireless sensor networks. In: *IEEE Communications Letters*, 2012, 16(11), pp. 1896–1899.
66. DE GANTE, A, ASLAN, M, MATRAWY, A. Smart wireless sensor network management based on software-defined networking. In: *2014 27th Biennial symposium on communications*, 2014, pp. 71–75.
67. HAN, Z, & REN, W. A novel wireless sensor networks structure based on the SDN. In: *International Journal of Distributed Sensor Networks*, 2014, (7), pp. 1–7.
68. Galluccio, L, Milardo, S, Morabito, G, Palazzo, S. SDN-WISE: Design, prototyping and experimentation of a stateful SDN solution for Wireless Sensor networks. In: *2015 IEEE conference on computer communications (INFOCOM)*, 2015, pp. 513–521.
69. Assunção, M, Carpa, R, et. al. Designing and building SDN testbeds for energy-aware traffic engineering services. In: *Photonic Network Communications*, 2017, 34 (15), pp. 1-15.
70. Gupta, M, Singh, S. Greening of the internet. In: *Computer Communication Review*, 2003, 33(4), pp. 19–26.
71. VASIC, N, KOSTIC, D. Energy-aware traffic engineering. In: *1st International Conference on Energy-Efficient Computing and Networking*, 2010, 18(4), pp. 169–178.
72. GOIRI, I, KATSAK, W, Le, K, NGUYEN, T, BIANCHINI, R. Designing and managing data centers powered by renewable energy. In: *IEEE Micro*, 2014, 34(3), pp. 8–16.

73. Bawany, N, Shamsi, J, Salah, K. DDoS Attack Detection and Mitigation Using SDN: Methods, Practices, and Solutions. In: *Journal of Cryptology*, 2017, 42, pp. 425–441.
74. CCTV-based botnet used for DDoS attacks. [Online]. [Cited: 04.07.2017]. Available: https://www.ddosattacks.net/a-massive-botnet-of-cctv-cameras-involved-inferocious-ddos-attacks .
75. DDoS Attack on Bank of Greece Website. [Online]. [Cited: 04.07.2018]. Available: https://www.hackread.com/anonymous-ddos-attack-bank-greece-website-down.
76. WONG, F, TAN, C. A survey of trends in massive DDoS attacks and cloud-based mitigations. In: *International Journal of Network Security & Its Applications*, 2014, 6(3), pp. 57–71.
77. Designated router. [online] [Cited: 29.04.2020]. available: https://www.techopedia.com/definition/25105/designated-router.
78. HOSSAIN, MD, SHEIKH, M, et. al. Quality of Service in Software Defined Networking. In: *Global Journal of Computer Science and Technology*, 2018, 18(3), pp. 21-28.
79. MISHRA, S, MISHRA, S, KUMAR, N. Hashing algorithm: MD5. In: *International Journal for Scientific Research and Development*, 2013, 1(9), pp. 1931-1933.
80. GOWTHAMAN, A, MANICKAM, S. performance study of SHA-265 algorithm. In: *International Journal of Applied Engineering Research*, 2015, 10(4), pp. 10921-10932.
81. PUTHAL, D, MALIK, N, MOHANTY, S, KOUGIANOS, E, DAS, G. "Everything You Wanted to Know About the Blockchain: Its Promise, Components, Processes, and Problems. In: *IEEE Consumer Electronics Magazine*, 2018, 7(4), pp. 6-14.
82. AYESHA, I. SDN controllers' security issues: MS thesis. *University of Jyväskylä* - Finland, 2017.
83. DESHMUKH, R, et. al. Understanding DDoS Attack & Its Effect In Cloud Environment. In: *Procedia Computer Science*, 2015, 49 (1), pp. 202-210.
84. IQBAL, M, RIADI, I. Analysis of Security Virtual Private Network (VPN) Using OpenVPN. In: *International Journal of Cyber-Security and Digital Forensics*, 2019, 8(1), pp. 58-65.
85. CHAWLA, B, GUPTA, O, SAWHNEY, B. A Review on IPsec and SSL VPN. In: *International Journal of Scientific & Engineering Research*, 2014, 5(11), pp. 21-24.
86. JAHAN, S. Application Specific Tunneling Protocol Selection for Virtual Private Networks. In: *International Conference on Networking, Systems and Security*, 2017, (1), pp. 39-44.

87. SOBH, T, ALY, Y. Effective and Extensive Virtual private network. In: *Journal of Information Security*, 2011, 2(01), pp. 39-49.
88. JAHA, A, SHATWAN, F, ASHIBANI, M. Proper Virtual Private Network (VPN) Solution. In: *The Second International Conference on Next Generation Mobile Applications, Services, and Technologies*, 2008, pp. 309-314.
89. TAN, W, et. al. Analysis of RSA based on Quantitating key security strength. In: *Procedia Engineering*, 2011, 15, pp. 1340-1344.
90. UKWUOMA, H, HAMMAWA, M. optimized key generation for RSA encryption: thesis. *Ahmadu Bello University Zaria-Nigeria*, 2015.
91. SARMAH, S. Understanding Blockchain Technology. In: *Journal of Computer Science and Engineering*, 2018, 8(2), pp. 23-29.
92. PUTHAL, D, MOVANTS, S. Everything you wanted to know about Blockchain: its Promise, Components, Processes and Problems. In: *IEEE Consumer Electronics Magazine*, 2018, 7 (4), pp. 6-14.
93. GHOSH, J. The blockchain: Opportunities for Research in Information Systems and Information Technology, In: *Journal of Global Information Technology Management*, 2019, 22(4), pp.235–242.
94. EZE, K, et. al. Smart Contracts: A Primer. In: *Journal of Scientific and Engineering Research*, 2018, 5 (5), pp. 538-541.
95. LU, L, et. al. Supply Chain Management. In: *International Encyclopedia of The Social and Behavioral Sciences*, 2015, 23(2), pp. 709 -713.
96. Consortium blockchain. [Online]. 2019, [Cited: 26.07.2019] available: https://www.mycryptopedia.com/consortium-blockchain-explained/.
97. SHEIKH, H, et. al. Proof of Work Vs Proof of Stake: A Comparative analysis and an approach to Blockchain Consensus Mechanism. In: *International Journal for Research in Applied Science & Engineering Technology*, 2018, 6 (12), pp. 786-791.
98. ARTHUR, G, GHASSAN, K, SRDJAN, C. et. al, on the security and performance of proof of work blockchains. In: *The 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016, pp. 3-16
99. XU, J. Are blockchains immune to all malicious attacks, In: *Financial Innovation*, 2016, 2, pp. 1-9.
100. DEL RIO C. Use of distributed ledger technology by central banks: A review, In: *Scientific Engineering Journal*, 2017, 8(5), pp.1-13.

101. NGUYEN, C, et. al. Proof of Stake Consensus Mechanisms for Future Blockchain Networks: Fundamentals, Applications and Opportunities. In: *IEEE Access*, 2019, 7, pp. 85727- 85745.
102. How bitcoin mining works. [Online]. 2019 [Cited: 10.09.2019] available: https://www.coindesk.com/information/how-bitcoin-mining-works.
103. How blockchain will manage networks. [Online]. 2019, [Cited: 26.08.2019] available: https://www.networkworld.com/article/3356496/how-blockchain-will-manage-networks.html.
104. https://docsend.com/view/5zragmb. Marconi foundation. (2018). Marconi Protocol: white paper. 2019.
105. ZHOU, Y, et. al. A Petri Net-based method to verify the security of SDN. In: *IOP Conference Series: Earth and Environmental Science*, 2019, 1 (234), pp. 1-6.
106. POURVAHAB, M, GHOLAMHOSSEIN, E. An Efficient Forensics Architecture in Software-Defined Networking-IoT using Blockchain Technology. In: *IEEE access*, 2019, 7, pp. 1-17.
107. JIMIN, G. Research on Computer Network Security Evaluation Based on Neural Network. In: *4th International Conference on Machinery*, Materials and Computer, 2018, 150, pp. 665-670.
108. Xuan, S, Man, D, et. al. Mathematical Performance Evaluation Model for Mobile Network Firewall Based on Queuing. In: *Wireless Communications and Mobile Computing*, 2018, volume 2018, pp. 1-13.
109. AHMED, U, RAZA, I, et. al. Modelling cyber security for software-defined networks those grow strong when exposed to threats. In: *Journal of Reliable Intelligent Environments*, 2015, 1(2), pp. 123-146**.**
110. Petri Net. [Online]. 2019, [Cited: 11.10.2019] available: https://www.techfak.uni-bielefeld.de/~mchen/BioPNML/Intro/pnfaq.html.
111. ASSAF, G, et. al. Coloured fuzzy Petri nets for modelling and analysing membrane systems. In: *Biosystems*, 2022, 212, pp. 1-10.
112. ALMUTAIRI, L, SHETTY, S. Generalized Stochastic Petri Net Model Based Security Risk Assessment of Software Defined Networks. In: proc. of *IEEE Military Communications Conference*, 2017, pp. 545-550.
113. Guțuleac, E, Zaporojan, S, et. al. APPROXIMATE MAIN VALUE PERFORMANCE ANALYSIS OF COMPUTING PROCESS USING SHPN WITH FUZZY PARAMETERS. In: *Journal of Engineering Science*, 2020, 27(3), pp.111-133.

114. KARAKUS, M, DURRESI, A. Quality of Service (QoS) in Software Defined Networking SDN: A Survey. In: *Journal of Network and Computer Applications*, 2017, 80, pp. 200-218.
115. WHITMAN, M, MATTORD, H. Principles of Information Security (book). USA, 2011. 658 p. ISBN-13: 978-1-111-13821-9.
116. Dell Poweredge Servers Special Deals and Offers. [Online]. 2022, [Cited: 02.06.2022] available: https://www.dell.com/en-uk/work/shop/deals/enterprise-deals.
117. Cisco Switch Catalyst 1000. [Online]. 2022, [Cited: 02.06.2022] available: https://www.router-switch.com/cisco-catalyst-1000-switches-price.html.
118. **AMEEN, A.** Software - Defined Networks - A General Survey and Analysis. In: *Journal of Engineering and Science*, 2018, 25 (3), pp. 61-73.
119. **AMEEN, A.** The Using of SDN Technologies for Security Insurance of Computer Networks, In: proc. of *Technical-Scientific Conference of TUM*, 2019, Technical University of Moldova, 1, pp.1-4.
120. PERJU, V, **AMEEN, A.** State Security Assurance through the Creation of High-Protected Computer Networks, In Proc. *of the International Conference "Security strategic environment: trends and challenges SSETC-2019"*, 2021, Armed Forces Military Academy, pp. 101-109.
121. **AMEEN, A.** Leveraging Blockchain Technology to Assure Security of SDN. In: proc. of *International conference on Electronics Communications and computing*, 2020, Technical University of Moldova, pp. 1-12.
122. **AMEEN, A.** Making Cyber Space Networks a Safer Work Environment After Covid-19 Using Software-Defined Networks' Technologies, In: proc. of *International scientific conference-evolution of military science in the context of new threats to national and regional security*, 2020, Armed Forces Military Academy, volume 1, pp. 246-260.
123. **AMEEN, A,** GUŢULEAC, E. Petri Nets Modeling for SDN topologies. In: *Journal of Engineering Science*, 2021, 28(2), pp.79-90.
124. **AMEEN, A.** modeling proposed hybrid software-defined network controllers' topology by using Petri Nets system. In: *Studia Universitatis Moldaviae*, 2021, 2(142), pp. 40-50.
125. **AMEEN**, A. MODELING PROPOSED SDN PARALLEL TOPOLOGY AND EVALUATION OF ITS RELIABILITY. In: *Polish Journal of Science*, 2023, (66), pp. 47-56.
126. **AMEEN, A.** Assuring the SDN security by modeling and comparing SDN proposed topologies using Petri Nets. In: *Journal of engineering science*, 2021, 28(4), pp. 93-105.

127. PERJU, V, Mastac, Ion, **AMEEN, A.** Modern Military Command and Control Systems and their Security Ensuring based on the SDN Technology. In: *Polish Journal of Science*, 2023, (69), pp. 45-51.

128. **AMEEN, A.** Evaluation of the Computer Networks Security Level Based on Petri Nets & a Set of Parameters. In: *Polish Journal of Science*, 2023, (68), pp. 82-91.

129. PERJU, V, **AMEEN, A.** Security Assurance of State-of-the-Art Military Command-and-Control Systems Using the SDN-Based Technologies. In: Proc. of *International Scientific Conference "Republic of Moldova in the context of the new regional security architecture"*, 2022, Armed Forces Military Academy, 1, pp. 169-182.

130. PERJU, V, **AMEEN, A.** ASSESSING THE SECURITY OF MILITARY COMPUTER NETWORKS BASED ON THE PETRI NETS MODELING AND A SET OF PARAMETERS. In: Proc. of *International Scientific Conference "The Republic of Moldova in the context of the new regional security architecture"*, 2024, Armed Forces Military Academy, (to be published).

LIST OF PUBLICATIONS OF THE AUTHOR ON THE THEME OF THE THESIS

1. AMEEN, A. Software - Defined Networks. A General Survey and Analysis, In: *Journal of Engineering Science*, 2018, 25 (3), pp. 61-73. DOI: https://doi.org/10.5281/zenodo.2557306.
2. AMEEN, A. The Using of SDN Technologies for Security Insurance of Computer Networks, In: proc. of *Technical-Scientific Conference of TUM*, 2019, Technical University of Moldova, 1, pp.1-4. https://ibn.idsi.md/en/vizualizare_articol/84710n.
3. PERJU, V, AMEEN, A. State Security Assurance through the Creation of High-Protected Computer Networks, In proc. Of *the International Conference "Security Strategic Environment: Trends and Challenges SSETC-2019"*, 2021, Armed Forces Military Academy, pp. 101-109.
4. AMEEN, A. Leveraging Blockchain Technology to Assure Security of SDN, In: proc. of *International conference on Electronics Communications and computing*, 2020, Technical University of Moldova, pp. 1-12. DOI: https://doi.org/10.5281/zenodo.4288305.
5. AMEEN, A, GUŢULEAC, E. Petri Nets Modeling for SDN topologies. In: *Journal of Engineering Science*, 2021, 28(2), pp.79-90.
   DOI: https://doi.org/10.52326/jes.utm.2021.28(2).06.
6. AMEEN, A. modeling proposed hybrid software-defined network controllers' topology by using Petri Nets system. In: *Studia Universitatis Moldaviae*, 2021, 2(142), pp. 40-50. DOI: https://doi.org/10.5281/zenodo.5094689.
7. AMEEN, A. Making Cyber Space Networks a Safer Work Environment After Covid-19 Using Software-Defined Networks' Technologies, In: proc. of *International Scientific Conference-Evolution of Military Science in the Context of New Threats to National and Regional Security*, 2020, Armed Forces Military Academy, volume 1, pp. 246-260.
   https://ibn.idsi.md/vizualizare_articol/162231.
8. AMEEN, A. MODELING PROPOSED SDN PARALLEL TOPOLOGY AND EVALUATION OF ITS RELIABILITY. In: *Polish Journal of Science*, 2023, (66), pp. 47-56. https://doi.org/10.5281/zenodo.8337094.
9. AMEEN, A. Assuring the SDN security by modeling and comparing SDN proposed topologies using Petri Nets. In: *Journal of Engineering Science*, 2021, 28(4), pp. 93-105. https://doi.org/10.52326/jes.utm.2021.28(4).08.
10. PERJU, V, Mastac, Ion, AMEEN, A. Modern Military Command and Control Systems and their Security Ensuring based on the SDN Technology. In: *Polish Journal of Science*, 2023, (69), pp. 45-51. DOI: https://doi.org/10.5281/zenodo.10400324.
11. AMEEN, A. Evaluation of the Computer Networks Security Level Based on Petri Nets & a Set of Parameters. In: *Polish Journal of Science*, 2023, (68), pp. 82-91.

DOI: https://doi.org/10.5281/zenodo.10132797.

12. PERJU, V, AMEEN, A. Security Assurance of State-of-the-Art Military Command-and-Control Systems Using the SDN-Based Technologies. In: Proc. of *International Scientific Conference "Republic of Moldova in the context of the new regional security architecture"*, 2023, Armed Forces Military Academy, 1, pp. 169-182.
13. PERJU, V, AMEEN, A. ASSESSING THE SECURITY OF MILITARY COMPUTER NETWORKS BASED ON THE PETRI NETS MODELING AND A SET OF PARAMETERS. In: Proc. of *International Scientific Conference "The Republic of Moldova in the context of the new regional security architecture"*, 2024, Armed Forces Military Academy, (to be published).

ANNEXES

Annex 1. Data extracted that led to calculation of average number of tokens of Serial topology using Petri Nets and PIPE software's GSPN analysis module

GSPN Steady State Analysis Results

Set of Tangible States

	P1	P10	P11	P12	P13	P14	P2	P3	P4	P5	P6	P7	P8	P9
M0	1	0	0	1	0	0	0	0	0	1	0	0	1	0
M1	1	0	0	0	0	1	0	0	0	1	0	0	1	0
M2	1	1	0	1	0	0	0	0	0	1	0	0	0	0
M3	0	0	0	1	0	0	0	0	1	0	0	0	1	0
M4	1	1	0	0	0	1	0	0	0	1	0	0	0	0
M5	0	0	0	0	0	1	0	0	1	0	0	0	1	0
M6	0	1	1	1	0	0	0	0	0	1	0	0	0	0
M7	0	1	0	1	0	0	0	0	1	0	0	0	0	0
M8	1	0	0	1	0	0	0	0	1	0	0	0	0	1
M9	0	0	0	1	0	0	0	0	0	0	1	0	1	0
M10	1	0	0	0	1	0	0	0	1	0	0	0	1	0
M11	0	1	1	0	0	1	0	0	0	1	0	0	0	0
M12	0	1	0	0	0	1	0	0	1	0	0	0	0	0
M13	1	0	0	0	0	1	0	0	1	0	0	0	0	1
M14	0	0	0	0	0	1	0	0	0	0	1	0	1	0
M15	1	0	1	1	0	0	0	0	0	1	0	0	0	1
M16	0	1	0	1	0	0	0	0	0	0	1	0	0	0
M17	1	1	0	0	1	0	0	0	1	0	0	0	0	0
M18	1	0	0	1	0	0	0	0	0	0	1	0	0	1
M19	0	0	0	1	0	0	0	1	1	0	0	0	0	1
M20	1	0	0	0	1	0	0	0	0	0	1	0	1	0
M21	0	0	0	0	1	0	0	1	1	0	0	0	1	0
M22	1	0	1	0	0	1	0	0	0	1	0	0	0	1
M23	0	1	0	0	0	1	0	0	0	0	1	0	0	0
M24	1	0	0	0	0	1	0	0	0	0	1	0	0	1
M25	0	0	0	0	0	1	0	1	1	0	0	0	0	1
M26	0	0	1	1	0	0	0	0	1	0	0	0	0	1
M27	1	1	0	0	1	0	0	0	0	0	1	0	0	0
M28	0	1	0	0	1	0	0	1	1	0	0	0	0	0
M29	0	0	0	1	0	0	0	1	0	0	1	0	0	1
M30	1	0	0	0	1	0	0	1	1	0	0	0	0	1
M31	0	0	0	0	1	0	0	0	0	0	1	1	1	0
M32	0	0	0	0	1	0	0	1	0	0	1	0	1	0
M33	0	0	1	0	0	1	0	0	1	0	0	0	0	1
M34	0	0	0	0	0	1	0	1	0	0	1	0	0	1
M35	0	0	1	1	0	0	0	0	0	0	1	0	0	1
M36	0	1	0	0	1	0	0	1	0	0	1	0	0	0
M37	1	0	0	0	1	0	0	1	0	0	1	0	0	1
M38	0	0	0	0	1	0	1	1	1	0	0	0	0	1

M39	1	0	0	0	1	0	0	0	0	1	0	1	1	0
M40	0	1	0	0	1	0	0	0	0	0	1	1	0	0
M41	0	0	1	0	0	1	0	0	0	0	1	0	0	1
M42	1	1	0	0	1	0	0	0	0	1	0	1	0	0
M43	0	0	0	0	1	0	0	0	1	0	0	1	1	0
M44	0	1	0	0	1	0	0	0	1	0	0	1	0	0
M45	0	1	0	1	0	0	0	0	0	1	0	1	0	0
M46	0	1	0	0	0	1	0	0	0	1	0	1	0	0

Steady State Distribution of Tangible States

Marking	Value
M0	0.00379
M1	0.01923
M2	0.03702
M3	0.00063
M4	0.09531
M5	0.00397
M6	0.00617
M7	0.00383
M8	0.00994
M9	0.00617
M10	0.01349
M11	0.02691
M12	0.01386
M13	0.01962
M14	0.0284
M15	0.0037
M16	0.03333
M17	0.03046
M18	0.03024
M19	0.00249
M20	0.03433
M21	0.00337
M22	0.06246
M23	0.0252
M24	0.10346
M25	0.00737
M26	0.00093
M27	0.12332
M28	0.01128
M29	0.00083
M30	0.01634
M31	0.01202
M32	0.02563

M33	0.02113
M34	0.0041
M35	0.00031
M36	0.01845
M37	0.06535
M38	0.00817
M39	0.00687
M40	0.00858
M41	0.01072
M42	0.02747
M43	0.00172
M44	0.00515
M45	0.00458
M46	0.00229

Average Number of Tokens on a Place

Place	Number of Tokens
P1	0.70241
P10	0.47321
P11	0.13233
P12	0.14396
P13	0.412
P14	0.44404
P2	0.00817
P3	0.16337
P4	0.17375
P5	0.2958
P6	0.53046
P7	0.06867
P8	0.15964
P9	0.36715

Token Probability Density

	μ=0	μ=1
P1	0.29759	0.70241
P10	0.52679	0.47321
P11	0.86767	0.13233
P12	0.85604	0.14396
P13	0.588	0.412
P14	0.55596	0.44404
P2	0.99183	0.00817
P3	0.83663	0.16337
P4	0.82625	0.17375
P5	0.7042	0.2958

P6	0.46954	0.53046
P7	0.93133	0.06867
P8	0.84036	0.15964
P9	0.63285	0.36715

Throughput of Timed Transitions

Transition	Throughput
T1	0.07024
T10	0.01596
T11	0.01596
T12	0.00232
T15	0.00593
T16	0.0144
T17	0.0144
T18	0.00106
T19	0.00178
T4	0.00839
T5	0.01737
T6	0.01737
T9	0.00819

Sojourn times for tangible states

Marking	Value
M0	2.5
M1	5
M2	5
M3	1.11111
M4	10
M5	1.66667
M6	3.33333
M7	1.66667
M8	2.5
M9	1.66667
M10	2.5
M11	5
M12	2.5
M13	5
M14	2.5
M15	5
M16	2.5
M17	5
M18	5
M19	2
M20	5

M21	2
M22	10
M23	3.33333
M24	10
M25	3.33333
M26	2
M27	10
M28	3.33333
M29	3.33333
M30	5
M31	3.33333
M32	3.33333
M33	3.33333
M34	5
M35	3.33333
M36	5
M37	10
M38	5
M39	5
M40	5
M41	5
M42	10
M43	2
M44	3.33333
M45	3.33333
M46	5

State space exploration took 1.39s
Solving the steady state distribution took 0.059s
Total time was 1.946s

Annex 2. Data extracted that led to calculation of average number of tokens of Parallel topology using Petri Nets and PIPE software's GSPN analysis module

GSPN Steady State Analysis Results

There are 440 tangible states. Only a summary of the results will be displayed.
For complete results see C:\Users\alisa\OneDrive\Desktop\ЗНАНИЯ\petri nets\pipe\PIPEv4.3.0\Pipe\GSPN_Analysis.html

Average Number of Tokens on a Place

Place	Number of Tokens
P0	0.23658
P1	0
P10	0.10948
P11	0.78103
P12	0.32674
P13	0.32664
P14	0
P15	0.32669
P16	0.32663
P17	0.32674
P18	0.1093
P19	0.10933
P2	0.1093
P20	0.10948
P3	0.78139
P4	0.23664
P5	0
P6	0.10933
P7	0.78134
P8	0.23711
P9	0

Token Probability Density

	μ=0	μ=1	μ=2	μ=3
P0	0.7814	0.20133	0.01657	0.0007
P1	1	0	0	0
P10	0.89052	0.10948	0	0
P11	0.21897	0.78103	0	0
P12	0.72701	0.22359	0.04505	0.00435
P13	0.727	0.22369	0.04498	0.00433
P14	1	0	0	0
P15	0.72701	0.22364	0.04501	0.00434
P16	0.727	0.2237	0.04496	0.00434

P17	0.72701	0.2236	0.04504	0.00435
P18	0.8907	0.1093	0	0

P19	0.89067	0.10933	0	0
P2	0.8907	0.1093	0	0
P20	0.89052	0.10948	0	0
P3	0.21861	0.78139	0	0
P4	0.78134	0.20138	0.01658	0.0007
P5	1	0	0	0
P6	0.89067	0.10933	0	0
P7	0.21866	0.78134	0	0
P8	0.78103	0.20154	0.01671	0.00071
P9	1	0	0	0

Throughput of Timed Transitions

Transition	Throughput
T0	0.02186
T1	0.02187
T10	0.0273
T11	0.0273
T12	0.0273
T16	0.0273
T19	0.0273
T2	0.0219
T22	0.01093
T24	0.01093
T26	0.01095
T3	0.01093
T4	0.01093
T5	0.01095
T6	0.01093
T7	0.01095
T8	0.01093
T9	0.1

State space exploration took 3.051s
Solving the steady state distribution took 0.369s
Total time was 4.26s

Annex 3. Data extracted that led to calculation of average number of tokens of Hybrid topology using Petri Nets and PIPE software's GSPN analysis module

GSPN Steady State Analysis Results

There are 687 tangible states. Only a summary of the results will be displayed.
For complete results see C:\Users\alisa\OneDrive\Desktop\ЗНАНИЯ\petri nets\pipe\PIPEv4.3.0\Pipe\GSPN_Analysis.html

Average Number of Tokens on a Place

Place	Number of Tokens
P0	0.20446
P1	0
P10	0.09648
P11	0.71056
P12	0.30381
P13	0.30369
P14	0
P15	0.30375
P16	0.30368
P17	0.30381
P18	0.0963
P19	0.09632
P2	0.0963
P21	0.9037
P22	0.90368
P23	0.09648
P24	0.0963
P25	0.09632
P26	0.90352
P27	0.09648
P3	0.71111
P4	0.20452
P5	0
P6	0.09632
P7	0.71103
P8	0.20497
P9	0

Token Probability Density

	μ=0	μ=1	μ=2	μ=3
P0	0.80741	0.18107	0.01119	0.00034
P1	1	0	0	0

P10	0.90352	0.09648	0	0
P11	0.28944	0.71056	0	0
P12	0.74178	0.21586	0.03914	0.00323

P13	0.74177	0.21597	0.03905	0.00321
P14	1	0	0	0
P15	0.74178	0.21591	0.03909	0.00322
P16	0.74177	0.21598	0.03903	0.00321
P17	0.74178	0.21586	0.03913	0.00323
P18	0.9037	0.0963	0	0
P19	0.90368	0.09632	0	0
P2	0.9037	0.0963	0	0
P21	0.0963	0.9037	0	0
P22	0.09632	0.90368	0	0
P23	0.90352	0.09648	0	0
P24	0.9037	0.0963	0	0
P25	0.90368	0.09632	0	0
P26	0.09648	0.90352	0	0
P27	0.90352	0.09648	0	0
P3	0.28889	0.71111	0	0
P4	0.80735	0.18111	0.01119	0.00034
P5	1	0	0	0
P6	0.90368	0.09632	0	0
P7	0.28897	0.71103	0	0
P8	0.80704	0.1813	0.01131	0.00035
P9	1	0	0	0

Throughput of Timed Transitions

Transition	Throughput
T0	0.01926
T1	0.01926
T10	0.02582
T11	0.02582
T12	0.02582
T16	0.02582
T19	0.02582
T2	0.0193
T22	0.00963

T26	0.00965
T27	0.00963
T28	0.00963
T29	0.00963
T3	0.00963
T30	0.00965
T4	0.00963
T5	0.00965
T6	0.00963
T7	0.00965

T8	0.00963
T9	0.1

State space exploration took 4.047s
Solving the steady state distribution took 0.492s
Total time was 5.625s

Annex 4. Data extracted that led to calculation of average number of tokens of Ordinary topology using Petri Nets and PIPE software's GSPN analysis module

GSPN Steady State Analysis Results

Set of Tangible States

	P0	P1	P2	P3	P4	P5	P6	P7
M0	1	1	1	0	0	0	1	0
M1	1	1	0	0	1	0	0	0
M2	1	0	1	0	1	0	0	0
M3	0	1	1	0	1	0	0	0
M4	2	1	0	0	0	0	1	0
M5	1	2	0	0	0	0	1	0
M6	1	1	0	0	0	1	0	0
M7	1	0	0	1	1	0	0	0
M8	0	1	0	1	1	0	0	0
M9	2	0	1	0	0	0	1	0
M10	1	0	2	0	0	0	1	0
M11	1	0	1	0	0	1	0	0
M12	0	0	1	1	1	0	0	0
M13	0	2	1	0	0	0	1	0
M14	0	1	2	0	0	0	1	0
M15	0	1	1	0	0	1	0	0
M16	2	0	0	0	1	0	0	0
M17	0	2	0	0	1	0	0	0
M18	1	0	0	1	0	1	0	0
M19	0	1	0	1	0	1	0	0
M20	0	0	0	2	1	0	0	0
M21	0	0	2	0	1	0	0	0
M22	0	0	1	1	0	1	0	0
M23	3	0	0	0	0	0	1	0
M24	2	0	0	0	0	1	0	0
M25	0	3	0	0	0	0	1	0
M26	0	2	0	0	0	1	0	0
M27	0	0	0	2	0	1	0	0
M28	0	0	3	0	0	0	1	0
M29	0	0	2	0	0	1	0	0

Steady State Distribution of Tangible States

Marking	Value
M0	0
M1	0.00001
M2	0.00001
M3	0.00001
M4	0

M5	0

M6	0
M7	0.00001
M8	0.00001
M9	0
M10	0
M11	0
M12	0.00001
M13	0
M14	0
M15	0
M16	0
M17	0
M18	0.00002
M19	0.00002
M20	0.00002
M21	0
M22	0.00002
M23	0
M24	0
M25	0
M26	0
M27	0.99981
M28	0
M29	0

Average Number of Tokens on a Place

Place	Number of Tokens
P0	0.00009
P1	0.00009
P2	0.00009
P3	1.99975
P4	0.00008
P5	0.9999
P6	0.00002
P7	0

Token Probability Density

	μ=0	μ=1	μ=2	μ=3
P0	0.99992	0.00006	0.00001	0
P1	0.99992	0.00006	0.00001	0
P2	0.99992	0.00006	0.00001	0
P3	0.00007	0.0001	0.99982	0
P4	0.99992	0.00008	0	0
P5	0.0001	0.9999	0	0

P6	0.99998	0.00002	0	0
P7	1	0	0	0

Throughput of Timed Transitions

Transition	Throughput
T0	0.00001
T1	0.00001
T10	0.00001
T2	0.00001
T9	0.00001

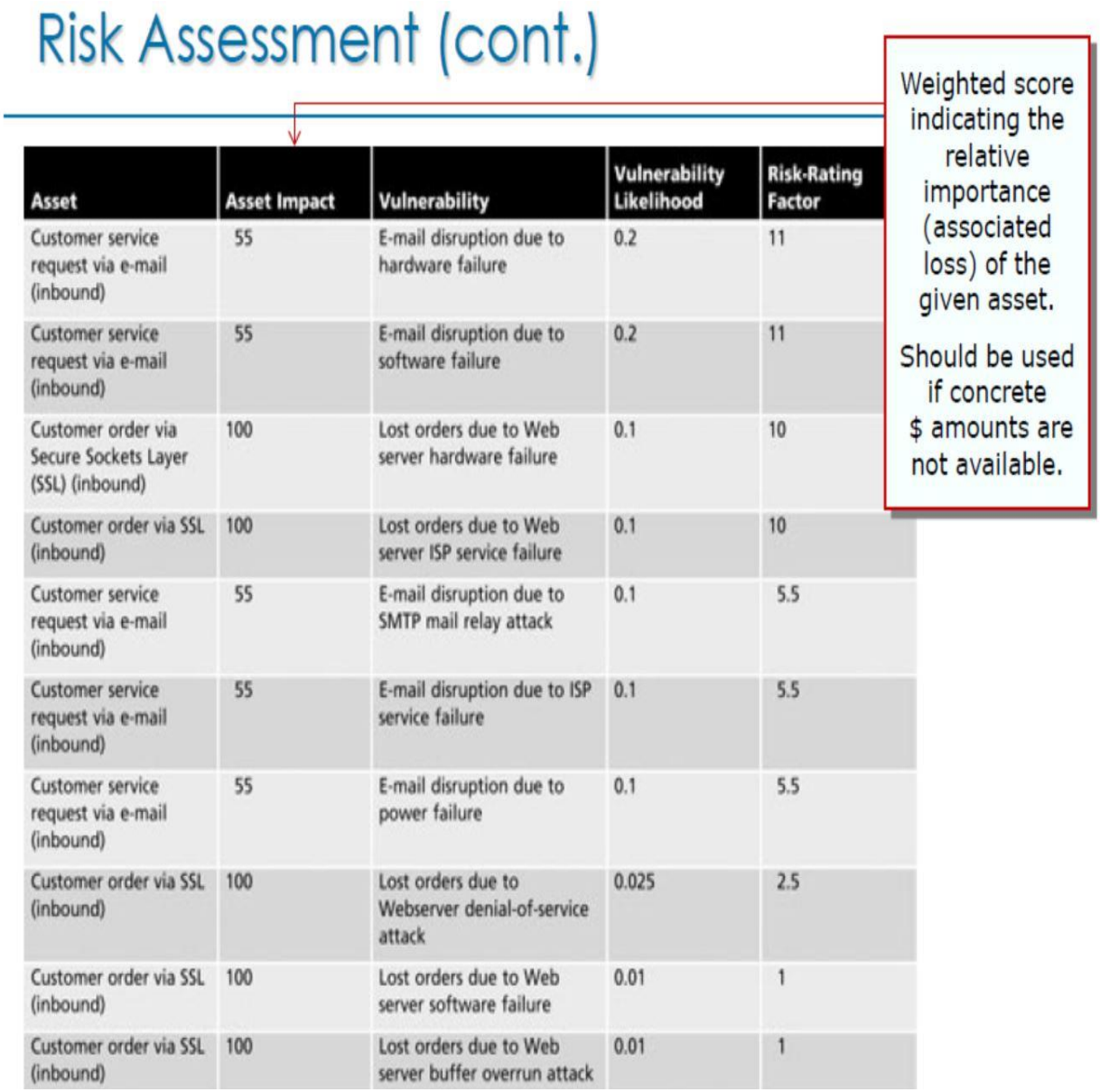
Sojourn times for tangible states

Marking	Value
M0	3.33333
M1	2.5
M2	2.5
M3	2.5
M4	5
M5	5
M6	5
M7	3.33333
M8	3.33333
M9	5
M10	5
M11	5
M12	3.33333
M13	5
M14	5
M15	5
M16	3.33333

M17	3.33333
M18	10
M19	10
M20	5
M21	3.33333
M22	10
M23	10
M24	10
M25	10
M26	10
M27	8
M28	10
M29	10

State space exploration took 0.256s
Solving the steady state distribution took 0.15s
Total time was 0.624s

Annex 5. Table showing the needed P and V values for calculating Risk Assessment in this research



Risk Assessment (cont.)

Weighted score indicating the relative importance (associated loss) of the given asset.

Should be used if concrete $ amounts are not available.

Asset	Asset Impact	Vulnerability	Vulnerability Likelihood	Risk-Rating Factor
Customer service request via e-mail (inbound)	55	E-mail disruption due to hardware failure	0.2	11
Customer service request via e-mail (inbound)	55	E-mail disruption due to software failure	0.2	11
Customer order via Secure Sockets Layer (SSL) (inbound)	100	Lost orders due to Web server hardware failure	0.1	10
Customer order via SSL (inbound)	100	Lost orders due to Web server ISP service failure	0.1	10
Customer service request via e-mail (inbound)	55	E-mail disruption due to SMTP mail relay attack	0.1	5.5
Customer service request via e-mail (inbound)	55	E-mail disruption due to ISP service failure	0.1	5.5
Customer service request via e-mail (inbound)	55	E-mail disruption due to power failure	0.1	5.5
Customer order via SSL (inbound)	100	Lost orders due to Webserver denial-of-service attack	0.025	2.5
Customer order via SSL (inbound)	100	Lost orders due to Web server software failure	0.01	1
Customer order via SSL (inbound)	100	Lost orders due to Web server buffer overrun attack	0.01	1

Annex 6. Certificate of the applied results at the Dekart Company

dekart
MAKE IT SECURE

dekart srl

Str. M.Kogalniceanu 85, mun. Chișinău Republica Moldova MD2009
Tel. +373 22 604290 fax. +373 22 808909 f/c 1002600037610
http: //www.dekart.com/, email: info@dekart.com

CERTIFICATE

By this Certificate is confirmed, that the **Mr. Ali Salman Hussein** research results on the topic "Security assurance of the computer networks based on software defined network technologies" were used in the company "Compania-Dekart S.R.L." investigations regarding the new approaches in information security.

The most interest presents the following.

1. SDN solution that provides the algorithms, methodologies, techniques and technologies which are proposed as a potential solution to enhance and assure the security of SDN environment to make it a safer and a more robust environment and to overcome the obstacles and security challenges in its way to facilitate the transition of computer networks management from classical structure to the SDN paradigm.

2. The proposed SDN controllers' topologies solution to overcome the centralization issue where the controllers interact in a specific way in every topology and after that comes a simple comparison of the effect of the algorithms in terms of their effect on each topology.

3. Defense Factor Equation to assess the security level of each of the proposed topologies in terms of their capabilities to DoS/DDoS attacks.

General director Valerii Anestiadi

Site: www.dekart.com
E-mail: info@dekart.com
Tel.: +373 022 604290

DECLARATION OF GUARANTOR

The undersigned hereby declares on the personal responsibility that materials presented in a doctoral thesis are the result of my own research and scientific achievements. Rationally understand that, otherwise, is to support the consequences in accordance with the legislation in force.

Ali Ameen

Signature:

Date:

CURRICULUM VITAE

Personal Data: Ali Ameen, date of birth 06.12.1988, Iraq, Baghdad.

Education:

2006-2010 Bachelor's degree in computer communications engineering, AL-Rafidain University, Baghdad-Iraq.

2013- 2015 Master's degree in Engineering and Engineering Activities, Free International University of Moldova, Moldova.

2017-2020 PhD student in computer communications engineering, Technical University of Moldova, Moldova.

Employment History:

01/2017–10/2017	Telecommunications engineer (IT): Al-Rajih Islamic Bank, Baghdad, Iraq.
11/2015–12/2016	marketing Agent, IT: Qatar Airways, Baghdad, Iraq.
08/2015–11/2015	Sales and marketing department manager: PHILIP MORRIS INTERNATIONAL- Marlboro, Baghdad, Iraq.
06/2015–08/2015	Telecommunications engineer (IT): Ericson, Baghdad, Iraq. Supervisor of Fiber optic design and mapping.
03/2015–06/2015	Computer Engineer: International Baghdad News Agency, Baghdad. Iraq.
11/2011–11/2012	Telecommunications engineer: SIDCCO (ministry of Industry & Minerals), Baghdad, Iraq.
06/2010–10/2011	Information Technician: Silk Road Group (SRG), Baghdad, Iraq.

Conferences:

1. Technical-scientific conference of TUM, 2019, presentation of report.
2. International conference, Security strategic environment: trends and challenges SSETC-2019, 23 May 2019, presentation of report.
3. ECCO conference, 23-26 October 2019, presentation of report.

Publications:

1. AMEEN, A. Software - Defined Networks. A General Survey and Analysis, In: *Journal of Engineering Science*, 2018, 25 (3), pp. 61-73. DOI: https://doi.org/10.5281/zenodo.2557306.
2. AMEEN, A. The Using of SDN Technologies for Security Insurance of Computer Networks, In: proc. of *Technical-Scientific Conference of TUM*, 2019, Technical University of Moldova, 1, pp.1-4. https://ibn.idsi.md/en/vizualizare_articol/84710n.
3. PERJU, V, AMEEN, A. State Security Assurance through the Creation of High-Protected Computer Networks, In proc. Of *the International Conference "Security Strategic*

Environment: Trends and Challenges SSETC-2019", 2021, Armed Forces Military Academy, pp. 101-109.

4. AMEEN, A. Leveraging Blockchain Technology to Assure Security of SDN, In: proc. of *International conference on Electronics Communications and computing*, 2020, Technical University of Moldova, pp. 1-12. DOI: https://doi.org/10.5281/zenodo.4288305.
5. AMEEN, A, GUŢULEAC, E. Petri Nets Modeling for SDN topologies. In: *Journal of Engineering Science*, 2021, 28(2), pp.79-90. DOI: https://doi.org/10.52326/jes.utm.2021.28(2).06.
6. AMEEN, A. modeling proposed hybrid software-defined network controllers' topology by using Petri Nets system. In: *Studia Universitatis Moldaviae*, 2021, 2(142), pp. 40-50. DOI: https://doi.org/10.5281/zenodo.5094689.
7. AMEEN, A. Making Cyber Space Networks a Safer Work Environment After Covid-19 Using Software-Defined Networks' Technologies, In: proc. of *International Scientific Conference-Evolution of Military Science in the Context of New Threats to National and Regional Security*, 2020, Armed Forces Military Academy, volume 1, pp. 246-260. https://ibn.idsi.md/vizualizare_articol/162231.
8. AMEEN, A. MODELING PROPOSED SDN PARALLEL TOPOLOGY AND EVALUATION OF ITS RELIABILITY. In: *Polish Journal of Science*, 2023, (66), pp. 47-56. https://doi.org/10.5281/zenodo.8337094.
9. AMEEN, A. Assuring the SDN security by modeling and comparing SDN proposed topologies using Petri Nets. In: *Journal of Engineering Science*, 2021, 28(4), pp. 93-105. https://doi.org/10.52326/jes.utm.2021.28(4).08.
10. PERJU, V, Mastac, Ion, AMEEN, A. Modern Military Command and Control Systems and their Security Ensuring based on the SDN Technology. In: *Polish Journal of Science*, 2023, (69), pp. 45-51. DOI: https://doi.org/10.5281/zenodo.10400324.
11. AMEEN, A. Evaluation of the Computer Networks Security Level Based on Petri Nets & a Set of Parameters. In: *Polish Journal of Science*, 2023, (68), pp. 82-91. DOI: https://doi.org/10.5281/zenodo.10132797.
12. PERJU, V, AMEEN, A. Security Assurance of State-of-the-Art Military Command-and-Control Systems Using the SDN-Based Technologies. In: Proc. of *International Scientific Conference "Republic of Moldova in the context of the new regional security architecture"*, 2023, Armed Forces Military Academy, 1, pp. 169-182.
13. PERJU, V, AMEEN, A. ASSESSING THE SECURITY OF MILITARY COMPUTER NETWORKS BASED ON THE PETRI NETS MODELING AND A SET OF PARAMETERS. In: Proc. of *International Scientific Conference "The Republic of Moldova in the context of the new regional security architecture"*, 2024, Armed Forces Military Academy, (to be published).