

The Evaluation of the Effects of Network Attacks in Wireless Mesh Networks

Yuming Zhu

A thesis submitted to
Auckland University of Technology
in partial fulfilment of the requirements for the degree
of
Master of Computer and Information Sciences (MCIS)

2016

School of Engineering, Computer and Mathematical Sciences

Table of Contents

List of Figure:	4
List of Tables:	5
Attestation of Authorship.....	6
Acknowledgements.....	7
Abstract	8
Chapter 1 Introduction	9
1.1 Background.....	9
1.2 Motivation	11
1.3 Contribution	12
1.4 Thesis structure.....	13
Chapter 2 Background	16
2.1 Wireless Mesh Network.....	16
2.2 Wireless Mesh Network Attacks	18
2.3 A Review of Simulation Tools	22
2.3.1 OMNeT++	22
2.3.2 ns-2.....	25
2.3.3 ns-3.....	25
2.3.4 OPNET.....	26
2.4 Simulator comparison	27
2.5 Simulation model	28
2.5.1 Neta	28
2.5.2 Attack scenarios	34
2.6 Gephi.....	38
Chapter 3 Topology modelling and Metrics	40
3.1 Background.....	40
3.2 Design and comparison	40
3.2.1 Bus topology	40
3.2.2 Star topology	42
3.2.3 Ring topology	43
3.2.4 Mesh topology	44
3.3. Metrics of mesh topology.....	45
3.3.1 Average degree	45
3.3.2 Degree distribution.....	45
3.3.3 Weighted degree	45
3.3.4 Betweenness centrality.....	46
3.3.5 Closeness centrality	46
3.3.6 Eccentricity	47
3.3.7 Graph density	48
3.3.8 Connected component.....	49
3.3.9 Clustering coefficient.....	49

3.3.10 Eigenvector centrality	50
3.3.11 Path length	51
3.4 Summary of Metrics	51
3.5 Case study of Metrics	53
Chapter 4 Case Studies	55
4.1 Case study I	55
4.1.1 Simulation Scenario	56
4.1.2 Result Analysis	57
4.2 Case study II	65
4.2.1 Three Topologies	65
4.2.2 Result Analysis	67
4.3 Summary of case study:	82
Chapter 5 Conclusions and future work	84
5.1 Conclusion	84
5.2 Future work	87
References	89
Appendix	

List of Figure:

Figure 1.1 Thesis Structure	13
Figure 2.1- Comparison between an original node and its attacker in NETA.....	31
Figure 2.2- Structure of simple dropping attack scenario.....	35
Figure 2.3- Structure of SimpleSinkholeRoute attack scenario.....	37
Figure 2.4- Structure of SimpleSinkholeNoRoute attack scenario.....	38
Figure 3.1- Example of bus topology structure.....	41
Figure 3.2- Example of star topology structure	42
Figure 3.3- Example of ring topology structure.....	43
Figure 3.4 Example of mesh topology structure.....	44
Figure 3.5 Topology for metric test	53
Figure 4.1 Twelve nodes mesh topology for case study.....	56
Figure 4.2 Packet lost value of eleven attack positions in Case Study I.....	57
Figure 4.3- Dropping value summary for each node of Case Study I	60
Figure 4.4- Case Study I Topology imported GUI in to Gephi	61
Figure 4.5 Betweenness centrality distribution of DIY topology	61
Figure 4.6 Eccentricity distribution of DIY topology.....	62
Figure 4.7 Closeness centrality distribution of DIY topology.....	62
Figure 4.8 Structure of topology A: 29 Nodes and 68 links	65
Figure 4.9 Structure of topology B: 29 Nodes and 68 links	66
Figure 4.10 Structure of topology C: 29 Nodes and 57 links	66
Figure 4.11 Topology A with nodes label	67
Figure 4.12 Topology A Packet dropping count.....	68
Figure 4.13 Clustering coefficient and Eigenvector performance of Topology A	69
Figure 4.14 Topology B with nodes label.....	72
Figure 4.15 Topology B Packets dropping count	73
Figure 4.16 Clustering coefficient and Eigenvector performance of Topology B	74
Figure 4.17 Topology C with nodes label.....	77
Figure 4.18 Topology C Packet dropping count.....	78
Figure 4.19 Clustering coefficient and Eigenvector performance of Topology C	79

List of Tables:

Table 2.1 Comparison of simulation tools studied	28
Table 3.1 Summary of Metrics	52
Table 3.2 T Fives metric results.....	53
Table 4.1 Dropping value of randomly changing two nodes into attackers in Case Study I	59
Table 4.2 Metrics results of Case Study I topology	64
Table 4.3 Metric results of first fifteen nodes of topology A	70
Table 4.4 Metric results of rest fourteen nodes of topology A	71
Table 4.5 Metric results of first fifteen nodes of topology B.....	75
Table 4.6 Metric results of rest fourteen nodes of topology B	76
Table 4.7 Metric results of first fifteen nodes of topology C.....	80
Table 4.8 Metric results of rest fourteen nodes of topology C	81
Table 4.9 Five metric comparison of three topologies	82

Attestation of Authorship

“I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person (except where explicitly defined in the acknowledgements), nor material which to a substantial extent has been submitted for the award of any other degree or diploma of a university or other institution of higher learning.”

Signature of Candidate:

Acknowledgements

I would like to express my deep and sincere gratitude to my supervisor, Dr William Liu, of the Department of Computing and Mathematical Sciences, Auckland University of Technology. His wide knowledge and logical way of thinking have been of great value for me. His understanding, encouragement and personal guidance have provided a good basis for the present thesis. And also the associate professor Nurul Sarkar, his brilliant advice helps me a lot during the final thesis finalized.

I owe my loving thanks to my parents. Without their encouragement and understanding it would have been impossible for me to finish this work.

Abstract

Wireless mesh networks are becoming an attractive solution for providing low-cost Internet access citywide. However, network attack is a critical and destructive behaviour in such networks, and is an extensive problem worldwide.

This thesis aims to identify the behaviour of different network attacks and their defence mechanisms, as well as their effective measurements. It mainly focuses on different protocols and models to figure out how the attacks behave and also evaluates various attacks. By doing so, some sort of advance consolidate can be exert to the existing defence solutions to handle with the network attacks and strengthen the network system to better understand the characteristics of cyber attacks, and the impact of cyber attacks on a wireless network. First, a simple simulation is conducted to study the various network attacks on basic topology structure with varying nodes and links. The purpose of this experiment is to understand the characteristics of the attack itself, and to get some insight into the network topology structure.

Besides that, in this thesis, the real attacks in a wireless mesh network have been reviewed. Furthermore a simple attack model has been implemented by simulation in the OMNeT++. The purpose of this thesis is not only to do research on network attacks, but our ultimate aim is to have a better understanding of a network's defence against cyber-attacks by optimizing the network structure so we can increase the security level of the network structure.

Chapter 1

Introduction

1.1 Background

In recent years, many security risks have emerged due to the “open” channel, which makes a network vulnerable to many network attacks, especially in the MANET (Mobile Ad Hoc Network). Due to its mobility and having no fixed nodes, it is highly possible that a simple attack is easier for an attacker. Besides that, the reduction of security workforces in the field of MANETs also makes the networks increasingly exposed to danger [1, 2]. Too many research texts concentrate on the theoretical classification of security risks, but they are not really useful to build attack detection systems for networks. For those reasons, it is clear that the suppression of these vulnerabilities and creating defences against these potential attacks is essential.

Previous work has quantified attacks, such as Blackhole attack, Collision attack, Delay attack and Eavesdropping attack, in wireless mesh networks [4]. Some of these attacks may have similar behaviours but different motivations. But in general, these attacks in MANETs can be divided into two groups: the active and passive, or they could be classified as the external and internal. In the second chapter, we address those attacks and give a more detailed description of them.

One of the best ways to get close to these real network attacks is to put them into your own design topology as the attack scenarios by simulating and seeing what happened. The purpose of the simulation is to understand the true effect of a new protocol, mechanism, network service and attacks. Normally the simulations are used to test network protocols and some complex systems; by doing so, the simulation framework

can offer a good compromise between cost and complexity. When it comes to a large network such as the Internet, the role of simulation is increasingly obvious [7]. Besides that, the use of a simulation framework is not only focused on the understanding level for network or protocols. In some situations, the framework is also necessary to act as the attacker in order to test your own system's security level.

NETwork attacks (NETA) is a framework which is intended to simulate a huge amount of attacks in heterogeneous networks using OMNeT++ [9]. OMNeT++ is one of the most used simulators due to the huge amount of frameworks and simulation of a large and growing set of overlay networks for evaluation purposes. NETA is intended to become a useful framework for researchers focused on the field of network security. Its flexible design is appropriate for the implementation and evaluation of many types of attacks, doing it accurately for the benchmarking of current defence solutions under the same testing conditions, or for the development of new defence techniques. NETA is based on the INET framework, which provides precise implementations of many different protocols in the computer networking protocol stack, as well as models for mobility, battery consumption, channel errors, etc [48].

The general idea is to develop models in OMNeT++ implemented as new nodes which can strike attacks: attacker nodes. In order to do this, the attacks are managed by the so-called attack controllers. These controllers manage one or more modules of a NETA framework attack node by sending control messages [13, 14]. These messages are sent from attack controllers to specific modules that implement a modified behaviour for the attack. They are called hacked modules hereafter. For implementing this modified behaviour, these hacked modules are inherited or replicated from INET modules and conveniently modified to obey the orders of attack controllers.

The OMNeT++ 4.x Integrated Development Environment is based on the Eclipse platform, and extends it with new editors, views, wizards, and additional functionality

[16]. OMNET++ adds functionality for creating and configuring models (NED and ini files), performing batch executions, and analyzing simulation results, while Eclipse provides C++ editing, SVN/GIT integration, and other optional features (UML modelling, bug tracker integration, database access, etc.) via various open-source and commercial plug-ins. OMNET++ is an extensible, modular, component-based C++ simulation library and framework, primarily for building network simulators. "Network" is meant in a broader sense that includes wired and wireless communication networks, on-chip networks, queuing networks, and so on. Domain-specific functionality such as support for sensor networks, wireless ad-hoc networks, Internet protocols, performance modelling, photonic networks, etc., is provided by model frameworks, developed as independent projects. OMNET++ offers an Eclipse-based IDE, a graphical runtime environment, and a host of other tools [9]. There are extensions for real-time simulation, network emulation, alternative programming languages (Java, C#), database integration, System C integration, and several other functions.

1.2 Motivation

According to Sánchez-Casado's case study of evaluating the effects of Network attacks, several general ideas about the development model of the NETA framework were mentioned [27]. A node was applied to act as the attacker node which can attack, and also the attack controllers were in charge of managing the attacks. These controllers send the control messages to ensure the attack was executed; besides that, these messages also can be seen as the hacked modules which obey the orders from the hackers [2, 4]. There are two rules which should be followed in building the framework process. Firstly, every base framework which is used must not be modified. Secondly, try to modify the original code as little as possible.

In 2012, Dini and Tiloca undertook a study about attack simulation frameworks for wireless sensor networks [23], which indicated that it is necessary to build an attack

simulation framework to describe attacks and to do a quantitative evaluation based on their network behaviour and also their performance. Another simulation example is Riley's work in 2003 [26]; the simulator called Georgia Tech Network Simulator (GTNetS) includes simulation models of a number of popular protocols at the application, transport, network, and link layer.

A black hole attack in a mobile ad hoc network was simulated with ns-2 in Al-Shurman, Yoo, & Park's work in 2004, which used the AODV source code for the network simulator [3]. Solutions like finding more routes to the destination and verifying the authenticity of the node were mentioned. Also, another article indicates that the mobile ad hoc network (MANET) is particularly vulnerable because of its fundamental characteristics like open medium and dynamic topology [25]. So there will be many security problems in wireless MANETs that need to be solved.

To better understand the characteristics of cyber attacks and the impact of cyber attacks on a wireless network, first, we planned to simulate some different network attacks addresses on the basic topology structure, and after that we came back to test the structure by adjusting the nodes or links of topology.

1.3 Contribution

This thesis focuses on the evaluation of the effect of network attacks on wireless mesh networks. We conducted different attacks as well as altered different structures of topologies which resulted in significant new knowledge. These contributions can be summarised as follows:

Firstly, by studying the basic attack simulation packet under NETA, we launched the attack on our own design topology and successfully modified some original NETA simulation parameters to make our experiment more thorough than the example of the simulation packet itself [2]. In the dropping attack simulation packet, as well as the delay attack simulation packet, we increased the number of nodes from three to

twelve as well as the connection links which from two to twenty. The first part of the case study will discuss the changes and describe this process in detail.

Secondly, we used the first part's result to speculate about the most effective parameter and apply three more complex topologies which have similar features in different aspects. After that, we successfully imported these three topologies into the Gephi (a topology analysis tool) which gave us more convincing metrics results [10]. By analysing the data, we confirmed our hypothesis that the high value betweenness centrality node as well as the lower value of clustering coefficient nodes is the critical access points in the whole topology which can result in more serious damage than others when they turn into the compromised nodes. And there is one metric that showed the score of a single node which marked the importance of each node in the whole topology structure.

1.4 Thesis structure

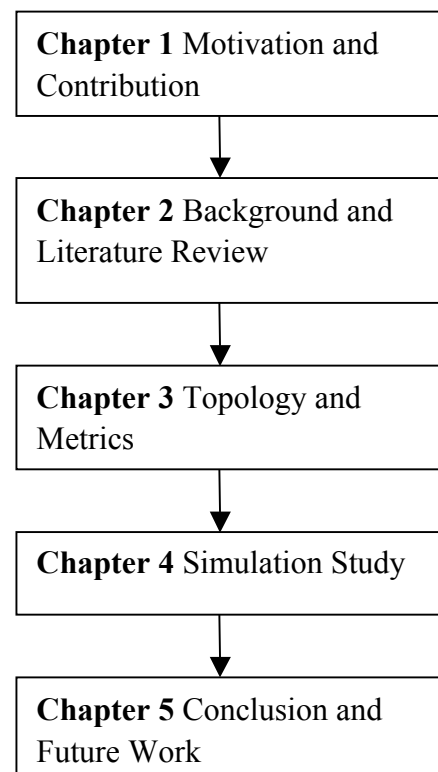


Figure 1.1 Thesis Structure

This research required thorough knowledge in the area of network attacks and defence. The initial stage of this research was to identify problems and the scope of the

experiment by carrying out an extensive review and assessment of the existing literature covering the area of network attacks and defence. Literature publications were gathered from the main credible sources of professional academic databases such as IEEE, ACM, Science Direct, and Springer. A qualitative survey on literature publications was carried out thoroughly by conducting critical analysis and comparison of the advantages and disadvantages of the existing solutions. Then, new theoretical and analytical models of network attacks and defences were proposed. The description of the problem solving procedure of the proposed model was created and validated using simulation tools. This was followed by detailed simulation and comparison analysis studies of the behaviours of network attacks and defences.

In Chapter 2, the thesis background material about this thesis topic, such as the background of wireless mesh networks and the attacks in wireless mesh networks, and also the concepts of simulations which include the simulator review and their comparison are considered. The simulation packet and graph analysis tool like NETA, Akaroa2 and Gephi will be discussed in this chapter too because of their importance for the simulation. Additionally, three attacks will be mentioned in the context of NETA; the attack scenarios in the NETA packet can be divided into three. The delay attack, the dropping attack and the sinkhole attack are the three main network attacks in the NETA simulation packet. The dropping attack was chosen as the experiment attack for the case study.

The third chapter is focused on the topology because it widely connects to the real network design and wireless mesh networks. Topologies that will be discussed in this section are: the bus topology, which is also known as the line topology; the star topology; the ring topology; and, the most well used topology nowadays, the mesh topology. After the comparison of each topology, the next section in this chapter will lay out several metrics of the mesh topology which are the key analysis parts for the results and also will be the content of the results section of the thesis. Nine metrics will be discussed in this section, and each one of them is the analysis component for

the topology evaluation in the case study chapter, which will be shown in the chapter on the results of the case study.

The fourth chapter is the case study and it is also the key contribution in this thesis. There are three parts in this chapter: the topologies part will show a self-design structure for the primary step simulation as case study one. Case study two is focus on the further study of three mesh topologies which have some similar characteristics. A summary of case study is located at part three.

The last chapter is the conclusion and ideas for future research which will lay out the results and conclude with the tables and graphs of the case study.

Chapter 2

Background

2.1 Wireless Mesh Network

Wireless Mesh Networks (WMNs) play an important role of offering a decent solution to building a wireless internet connection in a typical area. This technology can save a lot of money compared to the old-school WiFi, as WMNs can allow the whole network to develop in a promising way [1, 2]. In a regular Wifi network, plenty of wireless hot spots are needed for effective deployment, especially when further extending coverage of the network, as there are many more wireless hot spots that need to be added. This can cost a lot of extra money. In wireless mesh networks, it is possible to use only one wireless hot spot along with several wireless transit access points. These access points must rely on the wireless hot spots to deal with the traffic because there is no connection between these access points and those in the wired infrastructure [8]. However, the cost of transit access points is much cheaper than the cost of wireless hot spots, which means the wireless mesh network is quite economical. There are two situations where the WMNs are suitable to be installed [9]. The first one is that if there is no data cabling in the building for the wireless hot spots, and the second situation is that there is a need to build a temporary wireless network. However, the wireless mesh network is not suitable for large area deployment because of two reasons. The first one is because there might be interference when communication starts, which might be overcome by setting up multi-channel access points. The second reason is there are no security guarantees that the system is still safe when the deployment of WMNs slows down. In the following section, the

security challenges of WMNs will be listed, and also the possible solutions to secure the WMNs will be discussed.

In recent years, because of the increasing demands of internet communication and multimedia streaming, such as voice stream and video stream, there has been quite a huge development of wireless mesh network technology. This is mainly because a wireless mesh network can provide considerable support for the wired backbone for a reasonable price. Not only are WMNs more cost-effective, but they also have the ability to self-organize and self-heal, they can cover large areas, can be deployed quickly, and are easily installed. Besides those advantages, the WMNs are also effective in the large area application cases such as networking broadband and management of disasters [8, 13, 15 and 17]. They can also be set up without relying on the capacity of channels by using multi-hop communications. There are three components in a typical wireless mesh network: the mesh routers, the mesh clients, and the internet gateways which are known as the access points. The internet gateway is used as a bridge connecting the wired network to the top level of the hierarchy. The mesh router, however, is nothing more than a way to help the access points inter-connect by using their wireless links. The mesh route manages the traffic of mesh clients to the internet gateways, and the mesh clients are connected to their neighbour mesh routers with the multi-hop fashion.

The study of wireless mesh networks has been popular in recent years, and most studies address the protocol for routing purposes. However, the security issues of wireless mesh network, which are the critical parts for the new application of wireless mesh networks, is not as well addressed. There is a significant need for privacy protection and security mechanisms in this domain. In general, the wireless mesh network aims to help large quantities of data transmit between individual wireless transceivers like nodes. These individual nodes are cheap, reliable and resilient; the only duty for them is to forward the message to their next node. Because of this, these nodes make it possible to transmit information over large distances, even over rough terrain. The wireless mesh network itself can be very reliable because each node is connected to its neighbour nodes, and when one node is taken down for some reason,

the neighbouring nodes will carry on forwarding information by find another possible way to go. This is called the self-organizing feature. By adding extra nodes, the total capacity of a wireless mesh network can be expanded. It can use in mobile devices too. The message will be transferred from the initial device to the final destination by the shortest path, and that is why the wireless mesh network can be much cheaper than the classic network.

There are several security issues that need to be addressed in wireless mesh networks. First, the links in the wireless mesh network can be actively and passively attacked. Taking the passive attacks as an example, it could compromise confidentiality which could result in corrupted availability and authentication. Beside the links, the nodes in the wireless mesh network can be easily compromised from attacks inside and outside of the network because of a lack physical protection. Finally, because of the changeable feature of nodes, also known as self-organizing, the trust relationships between nodes can change regularly too.

The second security challenge is about the multi-hop routing. The attacker can insert false routing messages or alter the replicated node by changing its state or initiate a DoS attack to affect the routing part of the wireless mesh network.

The third challenge is the fairness. In a wireless mesh network, the majority of nodes act as a message transferor which is highly reliant on the nodes surrounding the initial one [13]. If the manager increases the number of nodes between the sender and the destination node, the bandwidth sharing can be easily decreased. In order to overcome this threat, the optimal configuration can be an effective way to optimize the bandwidth in the wireless mesh network. By giving the computation power to the nodes, it is possible to address the scheduling to make sure the per-client fairness and expand the bandwidth utilization in the wireless mesh network.

2.2 Wireless Mesh Network Attacks

In this section, we will discuss some main attacks that occur in the wireless mesh network, such as DoS attack, blackhole/sinkhole attack, wormhole attack, and others

[29, 31]. Some of these attacks can be the bridge for others, like the wormhole attack. It can cooperate with other attacks like blackhole and HELLO flooding attacks, which can cause significant damage to the topology, and significantly reduce the transfer performance. A summaries of the attacks is given below.

Denial of Service attack, also known as the DoS attack, is launched by intentional failure or external malicious behaviour. Normally when a DoS attack happens, the centralized resource is flooded during the operation so that the system is no longer able to perform actions correctly [21, 22 and 23]. In a DoS attack, there is an even more severe threat called distributed DoS attack, which is caused by several compromised nodes. These nodes are located inside the same network and collude together in order to slow down the network and cause some serious damage. For example, SYN flooding is one type of DoS attack. Denial of service attack is a common problem on the Internet nowadays. Not only the individual Internet user, but also governments and companies suffer when facing this attack. People use this attack to suppress others, like the people who rely on their web presence for doing business; a DoS attack can cause serious information loss or a huge loss of revenue. Even worst, the number of DoS attacks is still growing; there are on average 27.9 attacks every hour due to this problem [40, 41, 47]. However, there is a way to mitigate this type of attack. Normally, the DoS attack can reduce the usability of a network by sending a large quantity of data, known as flooding the packets [27]. In this case, the remote service can be the biggest victim of an attacker. By configuring the network, it can be an effective way to mitigate this attack. With the increasing number of attacks, the security challenge for the network will be serious because in the transport network as the big role in today's communication system, the network itself can allow a large number of users online at the same time.

Many things can be done with the internet, such as online shopping, trading and internet banking, and these activities need strong security protocols in order to protect users from various types of attacks. Most DoS attacks directly damage the individual

nodes and force them to accept their own service such as the web servers or providing some other straight access point [30, 32].

Besides the DoS attack, another serious attack type is called a routing attack, which can lead to several different attacks such as a wormhole attack and a sinkhole attack. For example, the routing table overflow attack intentionally creates non-functioning nodes to prevent other new paths, and it tries to push the protocol into a serious situation and it is can lead to a DoS attack as well [51, 55]. Another example is the wormhole attack which the receivers attack selectively to another location in the same network in order to act as a tunnel to resend those packets back to the network. Besides the wormhole attack, the blackhole or sinkhole attack is also a common routing attack. The malicious nodes announce they have the shortest path to the other node and try to intercept as many as they can [47]. Another attack named Byzantine attack which makes the compromised routing are not been scanned by insert their malicious nodes.

In a blackhole attack, the ultimate goal is to attract the traffic as much as possible from the whole topology to go through the compromised node [73]. Once the attack takes place in the centre of a topology, it can lead to many other attacks because no matter when the packets follow the nodes along, its path has many chances to tamper with application data. Normally, the blackhole attacks show their respect to the routing algorithm in order to give more attention to the surrounding nodes [34]. For example, one adversary could make an advertisement with a high quality route to a station. With the knowledge of end-to-end, some of the other protocols try to verify the quality of it [59, 62]. As long as there is powerful transmitter adversaries that can make the signal reach the destination station by a single hop, the neighbour will forward their packets right through this adversary [55]. At the same time, this adversary also creates a lot of attention by attracting all the traffic which goes through the compromised node [51]. By taking control of the traffic in the selected area, it is not difficult to modify the packets from all of the nodes which are located in the topology area. One of the biggest reasons why the sensor network is so susceptible to

the blackhole attack is that in the entire network, there is only one final destination or one station. Once the compromised node can create an effective path to the station, it is easy to attract most nodes surrounding it, which has no better path to choose [66, 68 and 69].

The impersonation attack also puts a wireless mesh network in grave danger, as once the authentication of parties is no longer being supported, it will give the compromised node a chance to join the network and send incorrect routing information, and even masquerade as some other trusted nodes [64, 65]. As long as the compromised node can get into the management system, it can change the configuration of the authorised users.

Another attack called Sybil attack can exist in the network with more than one identity to other nodes. The fault-tolerant multi-path routing and distributed storage can be extremely reduced by the Sybil attack. Besides the damage to the network, the Sybil attack is also the enemy of routing protocol too [36, 37]. The nodes are often told to exchange coordinate information with the surrounding nodes by the location-aware routing in order to route the geographically addressed packets efficiently. Normally, one node is willing to accept a single coordinate from their individual neighbour node, but the Sybil attack can be accepted more than once because of its multiple identities [75].

A wormhole attack is like there is a single node standing between two other nodes which prevents them from forwarding messages between them, but actually this single node is sending fake information to tell those two nodes they are actually the neighbour nodes [35]. This attack can work with the blackhole attack too. Once the compromised node announces it has a high quality path to the final station, the traffic is easily collected and goes through it as long as other path appears to be less attractive. And when the blackhole attack is on the other side of a wormhole attack, even the compromised node may stop sending information to the station. The exit of

the wormhole is always far from where the information is intended [70, 74]. By convincing two distant nodes that they are located next to each other, the wormhole attack can cooperate with many other types of attacks and it can be usually used in combination with forwarding or dropping.

The HELLO flood attack is another attack against the network, as the nodes are required to say HELLO to others in order to prove their neighbour relationship in many protocols. Once the node receives this kind of message, it can prove it's in the right position [38]. However, this can be fooled by a laptop-class attacker which has high-quality transmission power because it can convince each and every single node that the compromised node is their neighbour due to its powerful performance [76, 77 and 78]. The compromised nodes do not need to draw all the traffic in order to launch the HELLO attack [40]. It only needs to use the power to make sure every node received their packets, and also this attack can be broadcast along with the wormhole attack. Interestingly, the HELLO flooding attack is not like the usual flooding of sending the message over a multi-hop topology; a single hop is enough for the HELLO flooding attack to broadcast the message to all the nodes that act as the receivers [71, 72].

2.3 A Review of Simulation Tools

2.3.1 OMNeT++

According to study [23], OMNeT++ is open-source and open-architecture simulation environment which has powerful GUI support and simulation panel. It use language C++ and use it to provide the simulation class library. Beside the C++ language, another language is NED which the description of the topology of the network is written by the NED. These two languages can be loaded dynamically during the translated process by using a tool under OMNeT++ [12]. Based on the Eclipse

platform, the OMNeT++ simulation IDE extends it with new editors, views, wizards, and other functionality, which adds functionality in order to create and configure models (NED and INI files), also perform batch executions and analyze the simulation results. While Eclipse realizes C++ editing, SVN/GIT integration and other optional functions (UML modelling, bug-tracker integration, database access, etc.) by various open-source and commercial plug-ins, the environment will be instantly recognizable to those at home with the Eclipse platform.

By defining the Simulation Perspective, it is specifically designed for simulations. The Simulation Perspective contains a set of conveniently choosing views, which is arranged for easier creating NED, INI and MSG files [17]. This perspective is preferred for those people who are working with INI and NED files a lot, while other perspectives are aimed for different tasks as C++ development or debugging.

OMNeT++ is a kind of framework approach, which is not directly provide the simulation components for computer networks, queuing network, or other areas, it provides the basic machinery and tools to write this kind of simulation [11]. Support a specific application field of various simulation models and frameworks such as liquidity framework or INET framework. Developements of these models is completely independent of the OMNeT++, and follow their own release cycles [24]. Simulation model for the first time since its launch, the development of various individuals and research groups several areas, including: wireless and private network, sensor network, IP and IPv6 network, MPLS, wireless channel, peer-to-peer networks, storage area network (SAN), optical networks, queuing network, file systems, high speed interconnect (InfiniBand), and so on. Simulation model of port in real life like Linux quagga routing protocol implementation daemon or BSD TCP/IP stack, others have directly to OMNeT++ written [6]. In this thesis, section below will discuss in detail the project. In addition to the university research organizations and non-profit research institutions, companies such as IBM, Intel, Cisco, Thales and wide used OMNeT++ in the commercial success or internal research project.

OMNeT++ is an open source, based on the component of modularization and open architecture of discrete event simulation environment [13, 14, and 16]. Academic and non-profit use is free of charge. The main application field is simulation of communication network, but its general and flexible architecture is it can be used in other areas of simulation of complex IT systems, queuing network or hardware architecture. OMNeT++ is not a network simulator, but the current popular network simulation platform in the scientific community as well as the industrial countries, which builds a large user community [19]. OMNeT++ runs on Linux and other UNIX-like systems and Windows XP, Win2K OS platform. OMNeT++ is consisted of (1) the graphical network editor: a graphical web editor which allows graphic topology architecture, network to create the file (NED) language (2) the kernel library: a simulation kernel library contains the object definitions used to create the topology (3) the command line interface: including graphical and command line interface simulation execution model (4) documents. There are two types of modules: simple and compound module. Composite module is a set of simple modules. These modules assembled into larger components and model USES an advanced network.

Large-scale networks are allowed. Modules communicate by message exchanges. For example, these messages may represent the queuing network packet communications network or work [26]. They from a simple module sent to another simple module, directly to the destination or in a predefined path. Information exchange through the gates and the connection OMNeT++ provides a possibility to modify the existing models or create a new object classes probably from basic object classes (module, door, connection, etc.) [28]. Module type can be stored in a separate file. This allows the user to set of existing module type and the component library. OMNeT++ achieves deterministic modelling formalism. But it also handles the continuous and discrete random variables to random model [20]. OMNeT++ has a separate window text output for each module. Scheduled news can be seen with the development of the

simulation, the Event - by - the Event. Perform the animation. The simulation results in the implementation of the graphical display.

2.3.2 ns-2

The ns-2 is developed by university of California, Berkeley, a public sphere event-driven network simulator. Part of the NS - 2 is currently brewing (virtual Internet test bed) project (fermentation). NS - 2 to simulate a small network. NS - 2 based on the three languages: TCL script simulation wrote, OTCL defines the simulation parameters, c + + realize the scheduler [39]. NS - 2 outputs can be produced: general trace file format, the non-aligned movement trace file format, personalized tracking files. Ns - 2 are cheap, complex scenes is very easy to test, can quickly get the results, and popular support platform and agreement. Main drawback is that the real-time system is too complicated model and scalability issues.

2.3.3 ns-3

Ns - 3 network simulator is a discrete event simulator mainly engaged in research and teaching. Ns - 3 project started in 2006, is an open source project. *ns-3* has been developed to provide an open, extended network simulation platform for research and education network. Briefly, NS-3 packet data networks are operating mode, and perform, and to provide an analogue simulation engine for users [67]. Some reasons to use NS-3 include performing research, more difficult or impossible to perform on a real system to study in a high degree of control, repeatable environment system behaviour and how the network works [61]. Users will notice that the available model set ns-3 focuses on how Internet protocols and network modelling work, but ns-3 is not limited to Internet-based systems; some users using the NS-3 in a non-Internet-based system modelling.

NS-2 has a more diversified than non-contributed modules NS-3, because of its long history. However, NS-3 in the study of more detailed models of several popular areas (including complex LTE and WiFi models), which supports the realization of the code range of high-fidelity model admits very wide [60, 61]. Users may be surprised to learn that the entire Linux network protocol stack node in ns-3 package, using direct code execution (DCE) framework. NS-2 models can sometimes be ported to ns-3, especially if they have been implemented in C++ [63].

2.3.4 OPNET

An overall framework is developed to simulate and evaluate performances of all kinds of wired and wireless communication networks and distributed systems, and can be implemented on Windows 2000, XP, Linux and Solaris platforms. The graphical tools are included for scenarios and models conception, scenarios simulation, data collection and data analysis. The OPNET simulation from within project includes a series of scenarios. This project is through the OPNET is also known as the central project editor interface [43, 44]. All of the available function may access from the editor. It provides an access to other editor, put forward function includes nodes and the process model to create and build a packet format, and create the filter and parameters. OPNET provides many additional features include a high level architecture (HLA) module, which allows communication between the various simulators. OPNET allow layered model by defining a set of sub models represent a subnet or of network [57]. Modelling is done in the modelling environment and it is composed of three areas. (1) Network domain. It defines the communication network topology to simulate. (2) The node domain. Defined domain instantiation of node network domain, namely elements connected to the network, can send and receive data (3) process area domain to describe the process of each module (processor or queue) user programmable. They perform process or task of visualization and analysis of the results and analysis tools and filter editor. Analysis tools are designed according to the results of simulation or a series of simulations. The project editor

shows the simulation results. OPNET model modularity is a critical infrastructure assets model bringing the possibility to define and add a new model to simulate a specific infrastructure [48]. New parameter can also be specified for model behaviours (which may occur in the infrastructure, management, processing, etc.). Infrastructure, which namely can distinguish between process models, describe the code or program module associated to a node or link. However, this tool is quite complex, especially in the specific components to be developed [53, 56].

2.4 Simulator comparison

The discussion includes the comparison with respect platform, interaction with other simulators, visualizes action capabilities and scalabilities of different network simulator. Each has its own characteristics and suitable for different cases. Simulator should be chosen according to the study of motivation. Researchers must consider different programming languages, which simulates driving events over the pros and cons of time, based on the component or object-oriented architecture, the complexity of the simulator, and functions with and without other design options the same. The NS-2 and the OMNeT++ open source side must be the best choice for research in most cases. NS-2 is the most popular simulator of academic research, but usually complicated by its critics [45, 50]. However, it is mainly made research community. The OMNeT++ is gaining popularity in the education and industrial sectors. Instead, OMNeT++ NS-2 has a well-designed and powerful simulation engine GUI. The following table is a summary of these simulators functions [52].

Table 2.1 Comparison of simulation tools studied

Feature	OMNeT++	Ns-2	Ns-3	Opnet
Language supported	C++	C++/OTCL	C++/OTCL	C++/jAVA
License	Open source	Open source	Open source	Commercial
GUI support	Good	Poor	Decent	Excellent
Time taken to learn	Moderate	Long	Long	Long
Platform	Linux, mac-os, unix	Unix, mac-os, Microsoft window cygwin	Unix, mac-os, Microsoft window cygwin, FreeBSD	C, C++, Opnet modeler software

2.5 Simulation model

2.5.1 Neta

Network attacks (NETA) is designed to simulate OMNeT++ heterogeneous network attacks using the framework. Since the in-depth study in the field of network security, within this framework, we try to provide a useful tool to promote the development of the estimated impact, damage, cyber attacks, and possible defences

[2,8]. This tool has the ability to directly prove a defence technology to prevent cyber attacks effectiveness (detection or response), or a different defence technology between many other applications. NETA is an INET2.1.0 basic framework. Because Universal and scalable architecture NETA, and a huge amount of attack could be implemented in each protocol layer. NETA trying to become the basic reference tool for simulated network attacks and implement. It provides free access to the research community under the GPL license.

NETA is scalable and provides a new and heterogeneous network attacks have developed a high degree of versatility. Its purpose is to save the development of attack used for testing purposes in the effort, resulting in network security field, as a useful tool for the research community [2]. Parallel Simulation is a way to divide up the simulation model and simulate different part of it processors [8].

Three main components of an attack are included in NETA framework: (i) attack controllers, (ii) control messages, and (iii) hacked modules.

(1) Attack Controllers

The controller is to control the onset of the attack execution module. They can be configured directly in the .ini file, they have the following properties: - attack type (string): The name used to identify attacks. It should label located NA.ned file corresponds. - Active (Boolean): It indicates whether an attack is in analog or inactive. - The startTime (double): In this attack simulation starts. - End time (double): In this attack stops. - Attack specific parameters: function-specific attacks based on different configuration parameters.

The processes implemented by an attack controller for attack A_i in an attacker node can be listed as:

1. To get involved in the attack A_i , perform different hacked module.
2. Activation message is sent at startup, and can contain configuration information of the node activation exploits hackers module?

3. To send a message to cancel the deactivation of hacking attack node module by the end of time.

(2) Control Messages

These messages are involved in the implementation of the attack from the attacker hacked module sends the controller. Activation and deactivation of the information required to transmit their attack. In addition, these messages contain configuration information needed to perform the attack. It remarks the control message is sent directly to a hacked module is very important. This is the realization of our design principles in Article 2 is the best choice: "To minimize hacked to modify the original code module."

(3) Hacked Modules

These are modified to attack module. For example, packet dropping attacks often need to modify the IP forwarding module one. Therefore, dropping the attack execution means NETA of IPv4 module, it appears as a black module has been modified. Please note that there is only one hacked each modification module, rather than the implementation of hacking attacks for each module. If two different attacks need to modify the same module, the module will only exist individually invasion. For example, as it will be shown that both the delay and drop attacks are related to the IPv4 module. Therefore, we need to perform two attacks a single invasion IPv4 module. The design by only including their attack control framework designed to improve flexibility, allowing more than one attack while performing, for example, delay and drop attacks can be triggered by the same node.

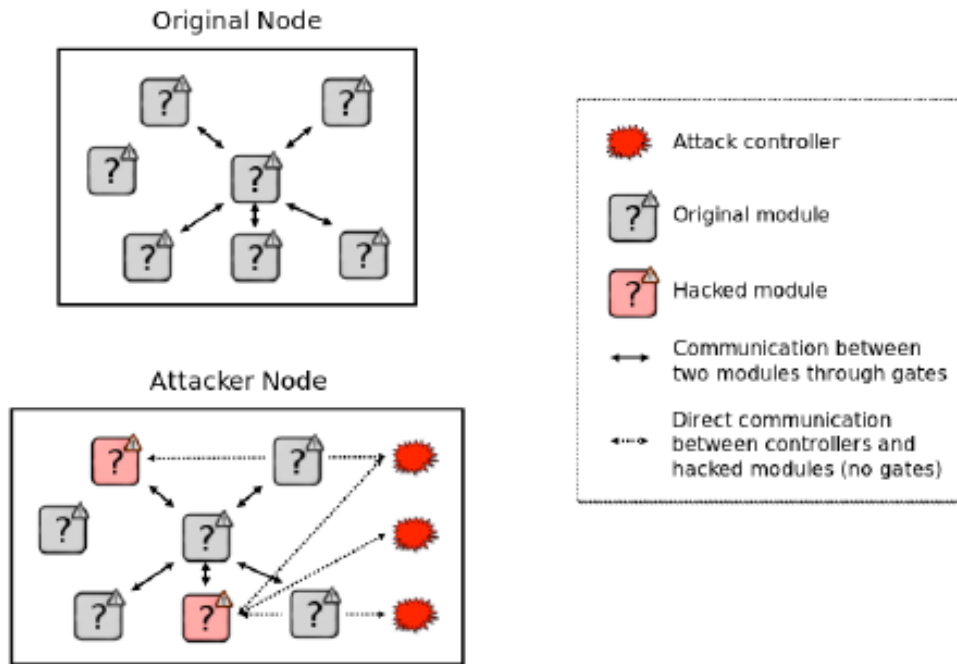


Figure 2.1- Comparison between an original node and its attacker in NETA

Fig. 2.1 shows the differences between a normal node and an attacker node. The normal node is composed of simple and compound modules communicating among them. The attacker node is composed of the same number of modules but now controller modules are added. In addition, some of the modules are replaced by hacked modules, in order to allow the execution of attack behaviours when triggered by attack controllers.

There are normally four components in each scenario's layout:

- `attacker[numDroppers] : NA_AttackerAdhocHost`

Attacker wireless ad hoc host. Modified from Wireless Host module. Redefined from INET framework and used as attacker host. To implement an attack you must include the corresponding attack controller module in the attacker host. This attack could be including N attacks controllers to implement several simultaneous attacks.

The current attack controllers included are:

NA-DroppingAttack (dropping attack controller)

NA_DelayAttack (dropping attack controller)

NA_SinkholeAttack (dropping attack controller)

- node[numHosts] : NA_AdhocHost

Wireless ad hoc host. Modified from WirelessHost module. Redefined from INET framework and used as normal host. It is used as a simple AdhocHost like in INET framework. These hosts never behave as attacker host. The only difference regards AdhocHost is that in this case they are using the MA framework architecture

- channelControl : ChannelControl

ChannelControl has exactly one instance in every network model that contains mobile or wireless nodes. This module gets informed about the location and movement of nodes, and determines which nodes are within communication or interference distance. This info is then used by the radio interfaces of nodes at transmissions and it must be named as "channelControl" inside the network.

- configurator : IPv4NetworkConfigurator

The module is assigned an IP address, and static routing IPv4 network. Number assigned to each interface IP address, subnets consider trying to take, can also optimize the routing tables generated by combining the routing entries. Configuration supports manual and automatic address assignment and combinations thereof. You can provide an address and subnet mask to partially uncertain, and trying to configure a single node on the LAN to automatically complete them within the same subnet. It also supports the manual route, shortest path and follows the automatic route. By default, the configuration adds a default route, if applicable (for example, the host) and performs routing based on the subnet. Hierarchical routing can be set up by using only a fraction of the number of nodes compared configuration items. Configurator also is optimized for large networks significantly reduce the size of the routing table, routing tables. Most of the features described above can be opened and closed using NED rotation parameters. For more information (interface addresses and network

mask templates, manuals, routing, etc.) may be in a separate XML file is configured to perform the entire network.

Akaroa is the original 2 (Akaroa 1) as described as the redesign and improved version. Like Akaroa 1 Akaroa 2 is written in an object-oriented C++ and Unix workstations running through LAN connection. It provides automatic general simulation program for parallel execution, the simulation output data analysis and simulation automatically stops automatically, to achieve the end result of the precision required.

Akaroa simulation package that can automatically generate and MRIP parallel program activation control process during the random simulation. Since then, we research activities in this area are concentrated in :(that is used to determine the accuracy of the estimation method for distributed development quality analysis) design Akaroa, automatic coverage analysis method is more efficient and user-friendly version by including other above-average estimated increase in Akaroa function. The interdisciplinary nature of the study, involving issues related to computer science and statistics.

However, division of it is very difficult due to its nature, and if the model does not divide easily, the relevant part of the cost of communication for a given simulation model simulation can be even longer. Akaroa 2 develops different methods [6-7] Application of multiple replicated in parallel or MRIP. Instead of dividing the simulation model, multiple independent instances of given models are executed simultaneously on different processors. The Akaroa 2 is generally used on the local network.

What are the principles to prevent any one group being distributed in the host simulation can reach each other over the Internet. Communication between a Akaroa2 simulation process requires only a very low bandwidth, and mostly one-way, so network latency less impact [9]. In theory, therefore, there should be differences in

performance between local or wide area network is unlikely. To test, simulate Akaroa2 run on PlanetLab are separated geographically research network node. Because the use of TCP / IP protocol to achieve this is very simple as the communication protocol. In addition to shooting to start the simulation time increased slightly, but no significant difference in simulation speed, as forecast.

2.5.2 Attack scenarios

Delay attack

In this attack, a malicious node delays may be observed for IP data packets. This can impact different QoS parameters (end-to-end delay, jitter, etc.), causing a unreliable network performance. The list of parameters in our implementation of the delay attack is:

*delay Attack Probability (double): Delayed data packet, the default probability is defined between 0 and 1, it is set to 0, which means that for the normal behaviour of the attacker node (no additional delay for any packet).

*delay Attack Value (double): Apply to a specific packet delay time. Note that this parameter can be specified by a statistical distribution. For this reason, it is defined as volatile, i.e., each time it is accessed been modified. By default, it follows with 0.1 seconds 1 second on average, the standard deviation of the normal distribution.

The original module is modified to strike the delay attack is also IPv4. We rename hacked module as NA IPv4.

Dropping attack

In the IP dropping attacks, this behaviour intentionally dumped node, has a certain probability of the received IP packets instead of forwarding them, disrupting normal

network operations. Depending on the application, it can open the network is much slower due to the presence of retransmission, so that the node waste more energy and resources to achieve our main parameters drop attacks:

*dropping Attack Probability (double): the probability of dropping a packet, defined between 0 and 1. By default, it is set to 0 which makes the attacker node to behave normally.

The original module is modified to strike the dropping attack is IPv4. We make the hacked module renamed as NA IPv4.

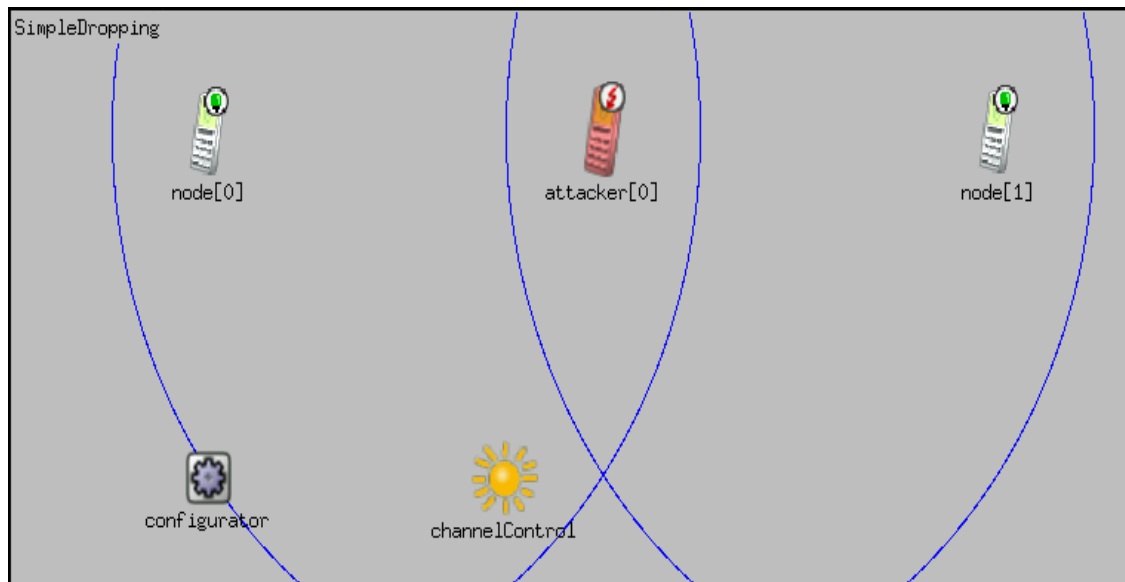


Figure 2.2- Structure of simple dropping attack scenario

Sinkhole attack

In sinkhole attacks, false routing information is sent by malicious nodes, it is claimed that it has one of the best route, and through it cause other nodes to route packets sent out. Here, the attacker forged the route reply (RREP), in order to attract traffic. Sinkhole attack parameter list is as follows:

-
- * `sinkholeAttackProbability` (double): the probability of answering a RREQ message with a fake route reply (RREP), defined between 0 and 1.
 - * `sinkOnlyWhenRouteInTable` (bool): if set to true, the sinkhole only sends fake RREP to requests for those that the attacker node has a valid route, for example, routes existing in its routing table. Otherwise (false value), the node sends fake RREP to any RREQ message arriving
 - * `seqnoAdded` (double): the fake sequence number generated by the attacker node. It is added to the sequence number differently each time, if it is specified as a statistical distribution. By default, it follows a uniform distribution with values between 20 and 30.
 - * `numHops` (int): the fake number of hops returned by the attacker. By default, it is set to 1, it shows that the attacker reaches the end of the communication in a single hop.

The original module is modified to strike the sinkhole attack is AODVUU. We make the related hacked module is renamed as NA AODVUU.

The sinkhole attack has two different cases, showing how the sinkhole attack works under two circumstances: whether the attacker knows the route to the final destination or not (`SimpleSinkholeRoute` and `SimpleSinkholeNoRoute` respectively). Through adding a value between 50 and 60, both scenarios the sink-hole node fakes the sequence number, which hops to 1.

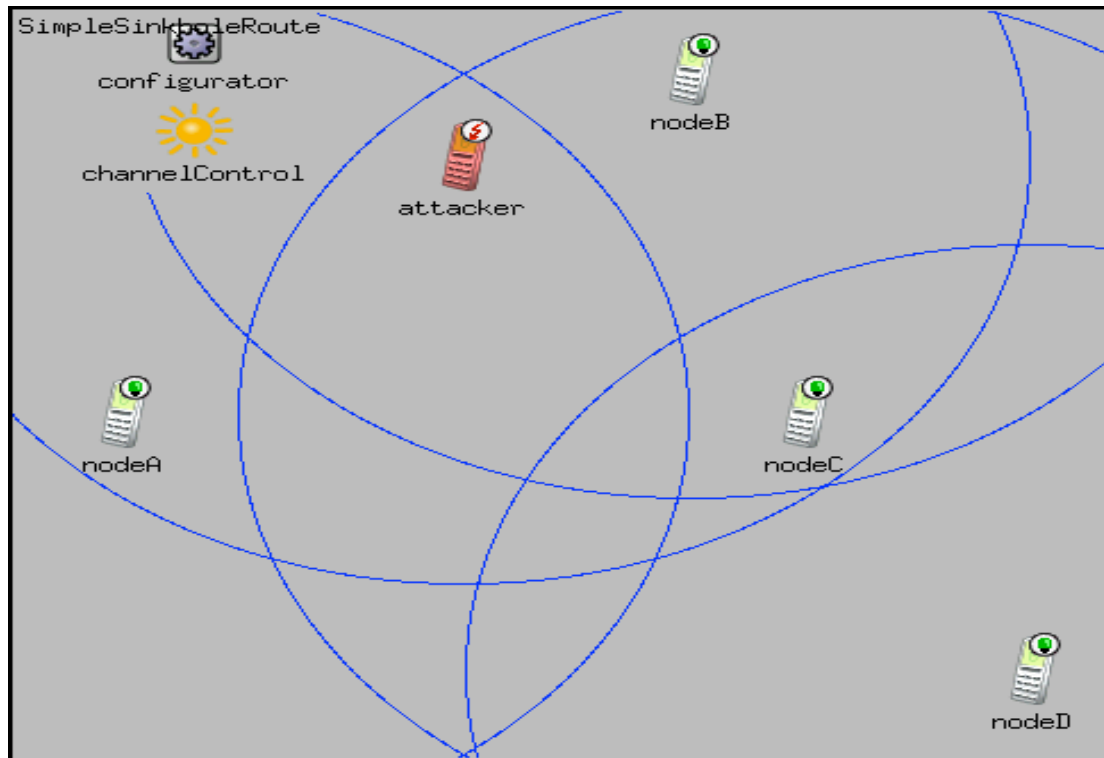


Figure 2.3- Structure of SimpleSinkholeRoute attack scenario

In the SimpleSinkholeRoute scenario (Fig. 2.3) there are 5 nodes: 4 normal NA AdhocHost and 1 NA AttackerAdhocHost. First, node A communicates with nodeD through the attacker between 0 and 1 s. Then, when the attacker has been known to join the route, the Node B is also a request for communication with the node D. Even know a hop routing node, Node B will choose, because of its reply to the false attack hop.

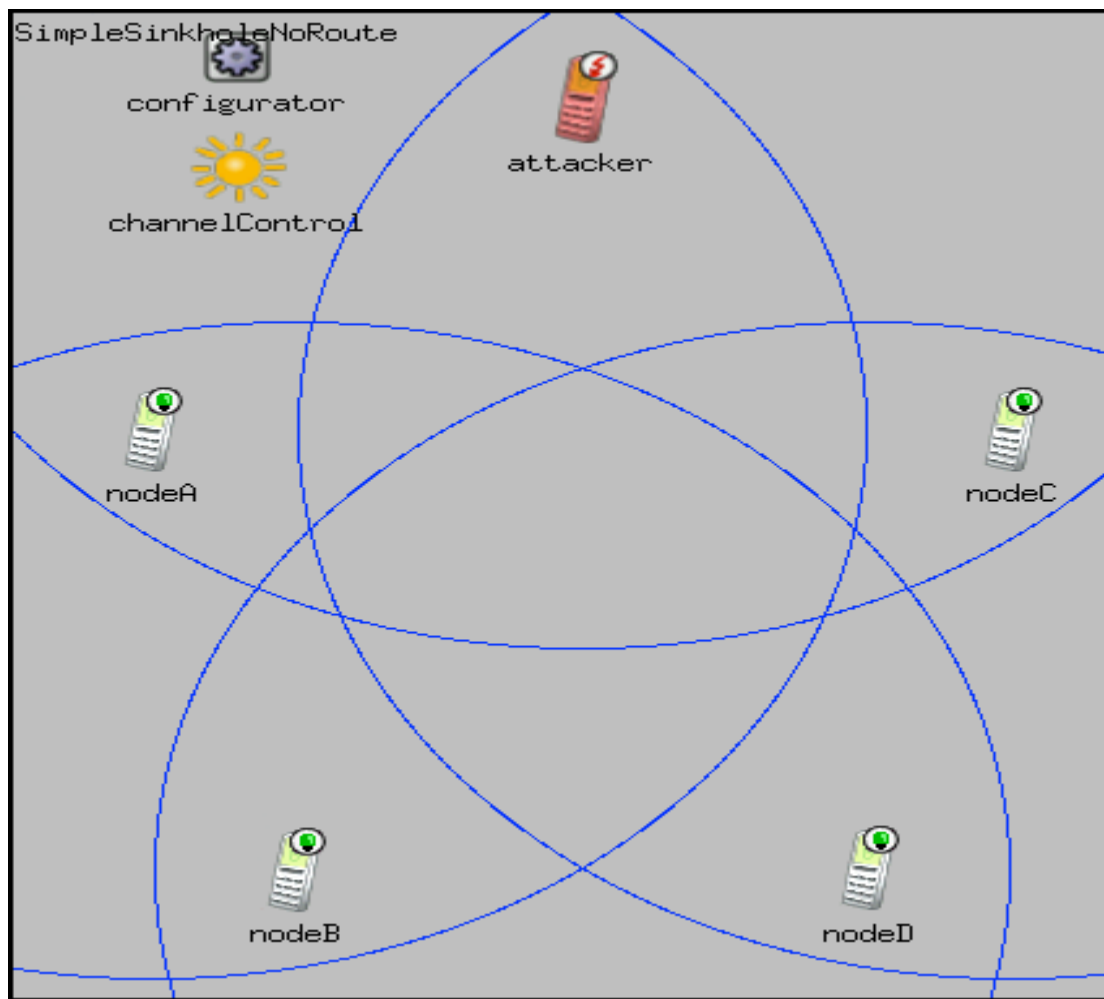


Figure 2.4- Structure of SimpleSinkholeNoRoute attack scenario

Fig. 2.4 shows the SimpleSinkholeNoRoute scenario, also with 4 NA AdhocHost nodes and 1 NA AttackerAdhocHost. Here, the attacker is selected as the next hop by node A because that the attacker sends a fake reply even though it does not know the route to the destination.

2.6 Gephi

The main function of Gephi is to provide a exploration platform for network topology with complex structures. As far as a topology concerned, a complete topology structure is made of different nodes, edges and relationship between them. For further

development, the Gephi can import, manipulate, filter and export topology by its developed module [10]. Besides that, in order to process multi-task, Gephi uses 3D rendering through a graphics card of a PC and the nodes can be added along with the images and there is no overlapping as well. Algorithm allows data to be moulded by the user, including involving speed, gravity, repulsion, automatic stabilization, inertial and sizing real-time motion. These algorithms are easy to choose and ensure that any user can benefit from the work of real-time functions [15]. These help to digest large amounts of data, such as the Semantic Web plug-in, which allows for a particular SPARQL, the query language, by searching for large files DBpedia, the database Wikipedia entries, including.

Using Gephi based on two popular items in connection with Google page ranking algorithm to find the republic of the 19th century's most influential novels and literature. In the letter, the Republic closer look shows the incredible range, it is possible to use Gephi. The site offers a list of letters and the relationship between the great thinkers in the history of Europe few.

In short, Gephi is a unique project in the history of the world. If history is to track the processing of information in the digital age it is an important need to find new and useful methods [18]. Like Gephi application allows a person by improving the graphics and data visualization (Bastien) their perception. If you use this application came to the forefront of important conclusions can be found. However, Gephi never in the history of the body, unless the effort into learning and networking, graphics and a variety of digital tools to achieve their full potential [46].

Chapter 3

Topology modelling and Metrics

3.1 Background

Network topology is map of the physical location of the computers' in a network linked by cables. Using the correct topology is very important because each topology has its own advantages and disadvantages. In this section, we will focus on the topology analysis among four main types: bus topology, also known as the line topology; star topology; ring topology; and mesh topology. The features of each topology will be listed, along with their advantages and disadvantages. After introducing each topology, the metrics for topology analysis will be discussed. Nine metrics will be clearly defined as it will also show in the results located in the case study chapter. They are: average degree, weighted degree, network diameter, graph density, connected component, clustering coefficient, eigenvector centrality, path length and Hyperlink-Induced Topic Search (HITS). In the network diameter, there are three parameters which belong to this metric: betweenness centrality, closeness centrality, and eccentricity. The betweenness centrality is the main parameter which is closely related to topology structure evaluation.

3.2 Design and comparison

3.2.1 Bus topology

As shown in Figure 3.1, computers are connected linearly through a single bus topology or more cables. A network utilizing a bus topology is called a "bus network",

which was the prototype of Ethernet networks. Ethernet 10Base2 (also known as thin-net) is used for bus topology.



Figure 3.1- Example of bus topology structure

As the cheapest way to connect computers for the development of a workgroup or departmental LAN, bus topology has the disadvantage that a single loose connection or cable break can bring down the entire LAN.

Termination is an important issue in bus networks. Electrical signals sent from the computer are free to travel the entire length of the cable. Without termination, when the signal reaches the end of the wire, it may bounce back up and travel back down the wire. When the signal echoes back and forth along an unterminated bus, it is called ringing. A terminator absorbs energy and stops reflections.

Advantages of the bus topologies:

- It is easy to use and understand
- It is also inexpensive
- It is easy to expand and join the cable network by allowing it to travel longer distances with a repeater that amplifies the signal

Disadvantages of bus topologies:

Because the network components do not coordinate with each other in order to allow time to transfer information, bus topology creates too much network traffic through a lot of slow computers. It is very difficult to solve this problem, because the cable breaks or loose connections can cause reflections and reduce overall functionality of the network.

3.2.2 Star topology

A star topology links the computers by individual cables to a central unit, usually a hub or switch, as in Figure 2. In a star topology, the network nodes are connected by points to a central node (also known as the central switching station; it is usually a hub or switch) for the transmission of information to the destination node by the central node. The central node performs a centralized communications control strategy, and therefore the central node is quite complex; its burden is much heavier than the nodes'. In a star network, in any two nodes, communication through the central node must be controlled.

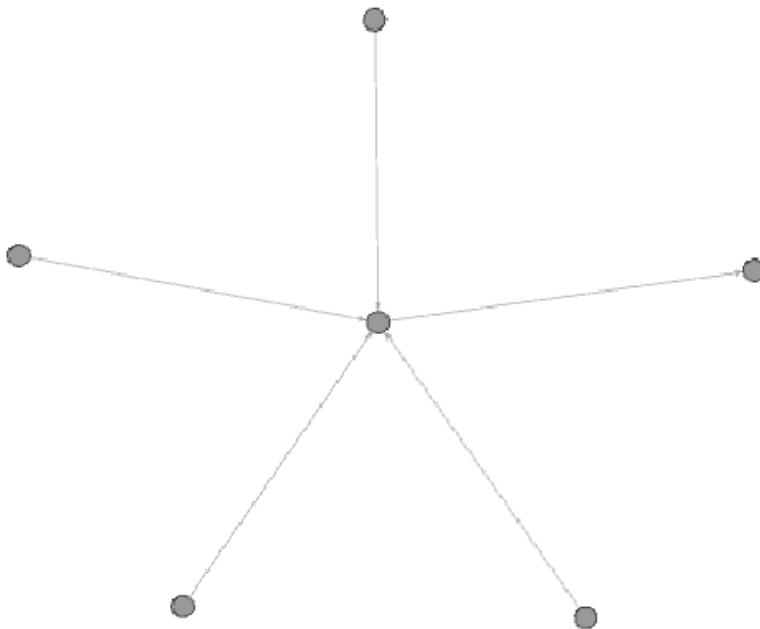


Figure 3.2 Example of star topology structure

Advantages of star topology are:

- The failure of a single point of attachment affects only one device and will not affect the entire network,
- The centralized networking equipment can reduce costs in the long run, And increase ease of network monitoring and management.
- It allows several cable types in the same network with a hub that can accommodate multiple cable types.

Disadvantages of star topology are:

- Because of central node burden, a "bottleneck" can form and cause failure, which affects the whole network
- It requires a lot of cables
- Installation is difficult
- The maintenance workload is high

3.2.3 Ring topology

Ring topology uses a common cable to form a closed loop. Each node is directly connected to the ring, so the transmission of information is in a certain direction along the ring from one node to another. A ring interface generally consists of a transmitter, a receiver, a controller, and the wire line receiver. In the ring topology, a transmission data control authority "token" surrounds a unidirectional transmission in the back, and each node must be received through a judgment node and issued to the node reception, otherwise data is sent back to the ring to continue down the path.

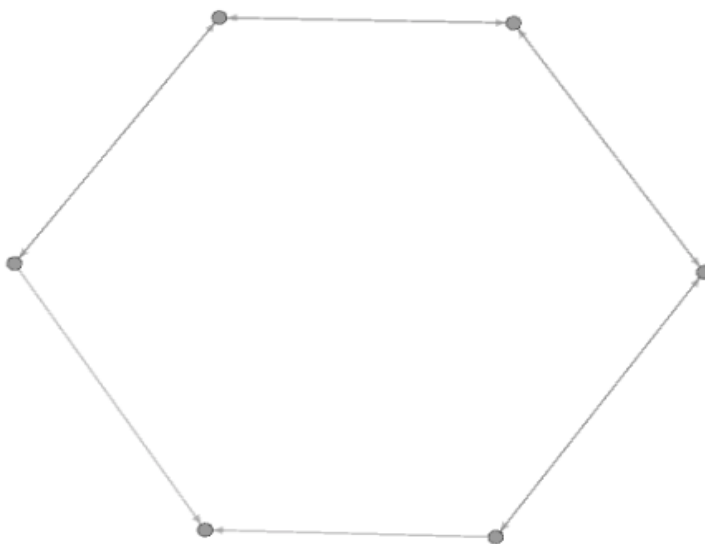


Figure 3.3- Example of ring topology structure

Advantages of ring topology are:

- All computers have equitable access to other parts of the network
- The network continues to function at a slower speed even after capacity is exceeded

Disadvantages of ring topology are:

- Node failure will cause the entire network to cease functioning
- It is difficult to troubleshoot
- The join node and the withdrawal process is complex

3.2.4 Mesh topology

In a mesh topology, each computer on the network has redundant data paths, as shown in Figure 3.4. This topology means that each node is interconnected mainly via a transmission line, and each node is connected to at least two other nodes. This results in high reliability, but its structure is complicated, costs to implement are high, and it is difficult to manage and maintain. Consequently, this topology is not often used for LAN.

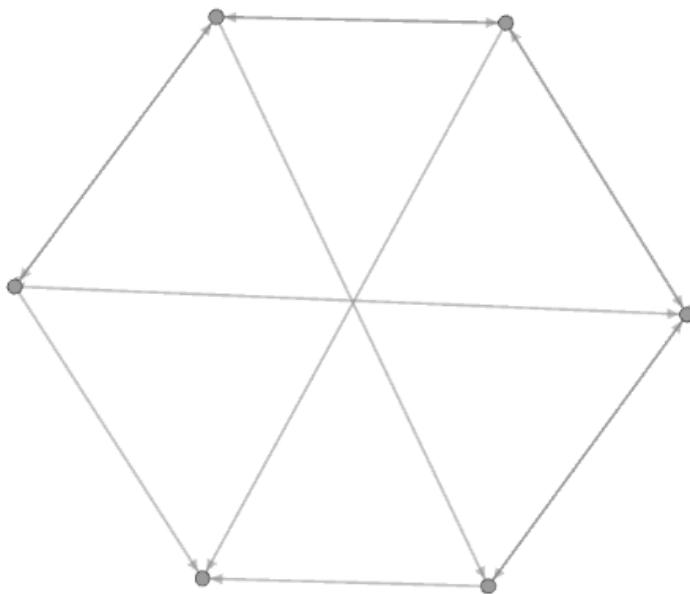


Figure 3.4 Example of mesh topology structure

3.3. Metrics of mesh topology

3.3.1 Average degree

In the research of graphics and networks, the network node degree is the number of other nodes connected to it and the degree of probability distribution in the whole network. The degree of a node in the network (sometimes called the error connection) is based on the number of connections from an edge node to other nodes [42]. In a network orientation, this means that the edge node points in one direction from one node to another node, and then there are two different node degrees: one degree is the number of incoming edges, and the other degree is the number of outgoing edges.

3.3.2 Degree distribution

The degree distribution of a network, referred to as the number of connections, is known as the edges [49, 54]. If the network is directly connected, then the edges travel in one direction from one to another. As long as the network is directed, there are two different degrees contained: the in-degree which indicates the incoming edges and the out-degree for the outgoing edges.

$$P(k) = \frac{n_x}{n}$$

The degree distribution $P(k)$ of a network can be defined in the network with the degree k . If there are n nodes in a network and n_x of them have degree k .

3.3.3 Weighted degree

The degree of a node is the number of relations (edges) it has. The weighted degree of a node is similar to the content of degree which is also based on the number of edges for a node, but is affected by the weight of each edge. It shows the sum of the weight

of the edges.

For example, a node with four edges with 1 weight ($1+1+1+1=4$) is equivalent to a node with two edges that weigh 2 ($2+2=4$), or a node with two edges that weighs 1 and one edge that weighs 2 ($1+1+2=4$), or a node with one edge that weighs 4 ($4=4$), and so on.

3.3.4 Betweenness centrality

Betweenness centrality indicates a node's centrality in a network. It is equivalent to the number of all the other nodes on the shortest path through all the vertices. This concept is widely used, including in computers and social networking, biotechnology, transportation and scientific cooperation.

The simplest way to measure this metric is to take the shortest path from all nodes to others on the topology and if one node is to go through each path, one point is added [58, 79]. So, when all the paths have gone through one from the other, there is a ranking of nodes based on the number of paths that went through it. A high number means a high betweenness centrality. This metric indicates how often a node is choosing a shortest path between two nodes in the network:

$$CB(v) = \sum_{u, w \in N, u \neq v \neq w} \frac{\sigma_{u,w}(v)}{\sigma_{u,w}}$$

$\sigma_{u,w}(v)$ is the number of shortest paths from the start node to the target node that pass through node v , and $\sigma_{u,w}$ is the total number of shortest paths from start node to target node.

3.3.5 Closeness centrality

Key nodes in the network core are close to the centre. Closeness centrality is defined as the inverse of the fair, which in turn, is the sum of all the other nodes' distance

[99,100 and 102]. If the distance between nodes in the disconnected parts of the network is unlimited, the measure cannot be applied with a trip member network.

This metric indicates how long it will take for messages from one node to reach other nodes in the topology.

$$CC(v) = \sum_{u \in N, u \neq v} \frac{\gamma(u, v)}{N}$$

The equation above shows the average shortest path from node u to all other nodes v . Because not all nodes are reachable from node u , it is appropriate to take the average of the paths.

3.3.6 Eccentricity

In mathematics, the eccentricity is a parameter associated with every conic section. It can be defined as a measure of how much the conic section deviates from being circular [80, 82]. For example, the eccentricity of a circle is zero. The eccentricity of an ellipse is between 0 and 1. The eccentricity of a parabola is 1, while the eccentricity of a hyperbola is above 1. Two conic sections are similar when they share the same eccentricity.

Any conic section is the locus of points with invariable distances to a point (the focus) and a line. That ratio of distances is eccentricity, commonly termed as e .

The eccentricity can also be defined from another perspective, which is the intersection of a plane and a double-napped cone associated with the conic section. If the cone is set up to have a vertical angle, the eccentricity is:

$$e = \frac{\sin \beta}{\sin \alpha}, 0 < \alpha < 90^\circ, 0 \leq \beta \leq 90^\circ$$

Where β is the angle between the plane and the horizontal, while α is the angle between the cone's slant generator and the horizontal. If the value of β is 0, then the plane section is a circle form, and if $\beta=\alpha$, then it is a parabola.

3.3.7 Graph density

Imagine everyone in the Facebook network is represented on a node graph. Users who expressed a shared friendship graph edges. When a user has increased the potential edge pattern density ratio (i.e., the user can have more friendships) what is increased with respect to a limited number of potential edges.

Graph density will be defined as the ratio of the number of edges and possible number of edges from an academic point of view [81, 83]. If I like the Facebook user services, they may calculate 100 as the edge for me, but if I only have 5 edges formed, then I have a relatively low pattern density. By increasing the number of edges of the pattern density, it can be increased and the interests of various densities occur from such an increase. For an undirected graph, the graph density can be defined as:

$$D = \frac{2|E|}{|V|(|V| - 1)}$$

For the directed graph, it is:

$$D = \frac{|E|}{|V|(|V| - 1)}$$

Where E is the number of edges and V is the number of vertices in the graph. The maximum number of edges is $|V|(|V|-1)$, so the maximal density is 1 and the minimal density is 0.

3.3.8 Connected component

In graph theory, since any two vertices are associated with each other by paths, a connected or unconnected component is an undirected graph and is not connected to additional vertices in the super graph [97, 98]. The number of connected components plays an important role in a graph. In topological graph theory, it can be interpreted as the zeroth Betti number of the graph. In graph theory, it equals the multiplicity of 0 as an eigenvalue of the Laplacian matrix of the graph [101, 104].

3.3.9 Clustering coefficient

Social networking has an important nature of clustering. People tend to have friends who are also friends with each other, resulting in the presence of the edge sets of many people, whereas people randomly selected from a series would have a much smaller number of edges between them. To measure a clustering of social networks, a common measure is the clustering coefficient [84, 85, 86 and 88]. The clustering coefficient is the largest cluster, when a network of disjoint faction is exactly zero, there is no clustering. Clustering in the network can be done in a variety of ways, but a common approach to do this is to check the triangle, that is, to check when the two sides share a node [103]. In a high clustering network, it is likely there will be a presence of the third edge, so that the three edges form a triangle. Although the clustering coefficient is often used, in fact, there are two variants of it which may have very different values.

Variant (1) – Clustering coefficient C_1 is defined as two incident edges are finished with a third probability triangle formation. The two sides share a vertex. In this case, we can check whether there is a third edge formed such that all three edges form a triangle. On the edge of such an event, it is completed by the third edge to form a triangle, which is equal to the ratio of the clustering coefficient (Variant 1).

Variant (2) – First, calculate the local clustering coefficient $c(u)$ for each node u as follows: Let $c(u)$ be the proportion of neighbours of u that are connected. In other words, $c(u)$ is the probability that two friends of u are friends with each other in a social network. Then the clustering coefficient (Variant 2) is defined as the mean of all local clustering coefficients in the network.

In graph theory, the clustering coefficient is a measure of the nodes of the graph that together measure the degree [87, 89 and 90]. There is evidence that in most real-world networks, particularly social networks, nodes tend to create a close relationship between the relatively high densities of population characteristics. The likelihood tends to establish a tie between two random nodes and then the average probability increases. The clustering coefficient for node u (undirected) is:

$$C(u) = \frac{2 \cdot |\{(v, w) \in E | (u, v) \in E \wedge (u, w) \in E\}|}{k_u \cdot (k_u - 1)}$$

The clustering coefficient of node u (directed) is:

$$C(u) = \frac{|\{(v, w) \in E | (u, v) \in E \wedge (u, w) \in E\}|}{k_u \cdot (k_u - 1)}$$

Where E is the set of edges, and k_u is the total degree of node u .

3.3.10 Eigenvector centrality

Eigenvector centrality is used to measure the influence of a node in a network as the node's importance in a network is based on a node's connections. It assigns relative scores to all nodes in the network from the idea that connections to high-scoring nodes contribute more to the score of the node than that to low-scoring nodes.

3.3.11 Path length

The average path length is a concept of network topology, and is defined as all possible steps shortest path network node for the average number of edge. It is a measure of the information on the web, or mass transfer efficiency [92, 94 and 96].

Along with the clustering coefficient and degree distribution, average path length is a robust measure of network topology. Some examples are: the average path length will guide you from one site to another place or you will have to pass exchanges, on average, with a complete stranger. It should not be confused with network diameter which is defined as one of the longest geodesic.

The negotiable network can be easily distinguished from one by average path length, which is complicated and inefficient, so a shorter path length is needed. However, the average path length is simply what the path length will most likely be. The network itself might have some very remotely connected nodes, and many nodes which are neighbours of each other [95].

3.4 Summary of Metrics

This section layout the summary of the metrics which show in the section above, it indicated their name, definition and some of those equations. However, the specific explanation of each equation is discussed in the related metric part in the former section.

Table 3.1 Summary of Metrics

Metric	Definition of the Metric	Equation of the Metric
Average degree	the average number of other nodes connected to it and the degree distribution of probability distribution in the whole network	
Degree distribution	the number of connections it has to other nodes over the whole network	$F(k) \approx k^{-\alpha}$
Weight degree	the degree of a node is the number of relation (edge) it has	
Betweenness centrality	the number of all other nodes in the shortest path through all the vertices	$CB(v) = \sum_{u, w \in N, u \neq v \neq w} \frac{\sigma_{u, w}(v)}{\sigma_{u, w}}$
Closeness centrality	the inverse of the fair, which, in turn, is the sum of all other nodes of the distance	$CC(v) = \sum_{u \in N, u \neq v} \frac{\gamma(u, v)}{N}$
Eccentricity	a measure of how much the conic section deviates from being circular	$e = \frac{\sin \beta}{\sin \alpha}, 0 < \alpha < 90^\circ, 0 \leq \beta \leq 90^\circ$
Graph density	measure how close the network is to complete	$D = \frac{2 E }{ V (V - 1)}$
Connected component	determine the number of connection components in the network	
Clustering coefficient	averages how nodes are embedded in their neighbourhood	$C(u) = \frac{ \{(v, w) \in E (u, v) \in E \wedge (u, w) \in E\} }{k_u \cdot (k_u - 1)}$
Eigenvector centrality	the influence of a node in a network	
Path length	all possible steps shortest path network node for the average number of edge	

3.5 Case study of Metrics

This section is shown how these metric perform on a simple topology which have four nodes and five links. The topology is tested with five metrics which contain calculate equations in section 3.4.

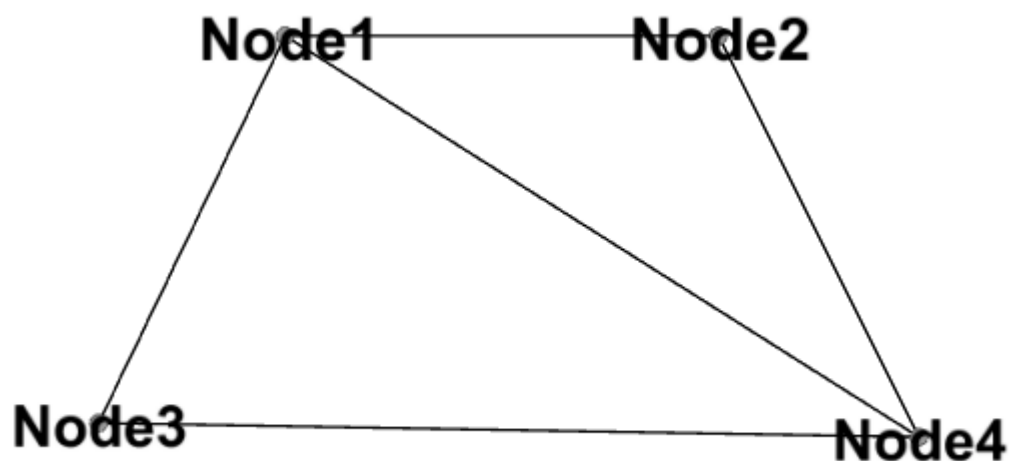


Figure 3.5 Topology for metric test

Table 3.2 T Fives metric results

Node ID	Eccentricity	Closeness Centrality	Betweenness Centrality	Clustering Coefficient	Graph Density
Node 1	1	1	0.5	0.667	0.833
Node 2	2	1.333	0	1	
Node 3	2	1.333	0	1	
Node 4	1	1	0.5	0.667	

As the graph density as an example, in this topology there are five edges which are node 1 to node 2, node 1 to node 3, node 1 to node 4, node 2 to node 4 and node 3 to

node 4. As the topology show above, there are four vertices. According to the equation of graph density:

$$D = \frac{2|E|}{|V|(|V| - 1)}$$

Where the E indicated number of edges and V represent number of vertices, the density of this topology is:

$$\frac{2|E|}{|V|(|V| - 1)} = \frac{2 * 5}{4(4 - 1)} \approx 0.833$$

Chapter 4

Case Studies

In this chapter, the main focus of our narrative is to assess the impact of topology structures on the attacks performance. In the first part of study, we will use simple topology for proof-of-concept: turn each node into the attack access point individually, then observe what different impact it has on the attack performance because their different locations. The comparison of results will be obtained and also the factors constituting the presumed result will be analyzed. Then we will study more complex topologies to validate these impacts. We use Gephi network analysis tool to measure the different characteristics of the topology. The aim is to see whether our demonstrate results and the hypothesis which we address before are match reasonable.

In second part of this chapter there are three more complex topologies to be compared in pairs. All of these three have the same numbers of nodes, and two of them have the same number of connection paths through layers of contrast and after that we conclude the characteristics of each one of them. Compared to the first experiment, this time we change the order of the conduction, we first import the three topologies into Gephi for parameter analysis and sum up the corresponding parameters, and then import the topologies into the OMNeT++ for attack simulation. The aim is to prove that the simulation result is associated with the corresponding topological metrics which have been selected for evaluation.

4.1 Case study I

The first case study focused on evaluating the position of an attacker in a multi-connect network, compared with the single line topology which is the attacker

and several nodes on the same line. The topology in the case study was more complex, as there were 12 nodes in the network and some of them connected with multiple neighbour nodes.

In order to evaluate how the position of the attacker has a different effect on the final output, we changed a normal node into an attacker node one by one, with the different locations of the attacker, in order to get a clear picture for the final output.

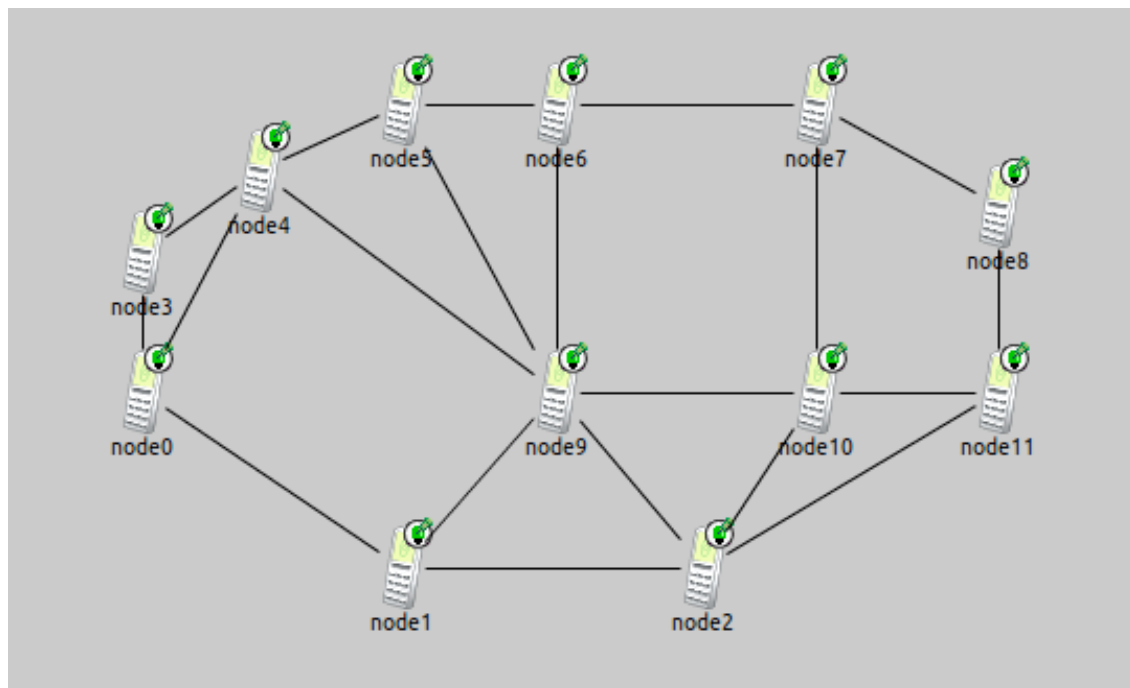


Figure 4.1 Twelve nodes mesh topology for Case Study I

4.1.1 Simulation Scenario

Figure 4.1 above shows the 12 nodes the packet received in a situation with no attacker in the network. The destination node was Node 11, and due to some nodes being connected to several node around it, there are also many back paths in the topology, and that is why there was a number of collisions during the packet delivery, which the blue bars stand for. The case study used the dropping attack as an example, with the dropping probability setting as 0.5 and simulation time 30s. The simulation

area was 1000*1000, message length was 512B and the routing protocol was AODVUU. The dropping attack node was intended to receive the packet from the formerly normal nodes but not forward it.

4.1.2 Result Analysis

Basically, the topology structure is shown in Figure 4.1 above, the node 11 is set as the destination node and each node will send packet to the destination node using their route, clearly, the node 9 connect with multiple neighbour nodes which is more than others. The next step is changing this normal node into the attacker node one by one and collects the result of those different cases. We collect all cases list them through a line chart.

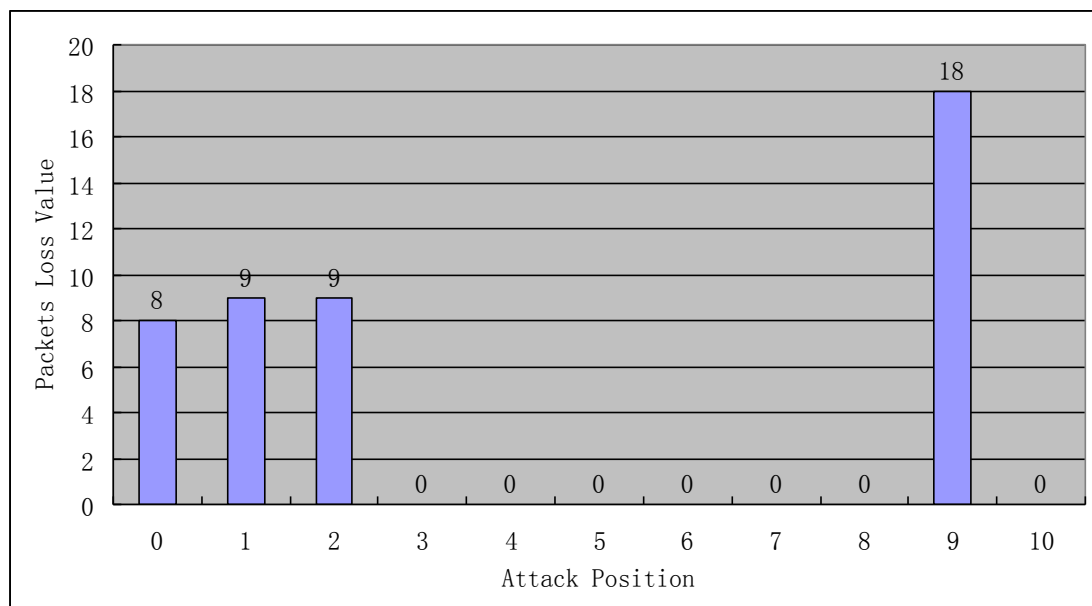


Figure 4.2 Packet lost value of eleven attack positions in Case Study I

According to Figure 4.2, four positions had their packets lost; they are position 0, position 1, position 2 and position 9 and they lost packets 8, 9, 9 and 18 respectively. Clearly, Node 9 is the most critical node in this topology, as when it changed into the attacker's node, there were 18 packets lost during the simulation process.

Node 1 and Node 2 both lost 9 packets and 8 packets were lost when Node 0 became the attacker. Looking back to the first graph where there was no attacker, the packets received shows how the nodes work in a normal situation in comparison. The green bars of Node 0, Node 1, Node 2 and Node 9 were higher than the average nodes, which mean there were more packets passing through those four nodes. But there were also unexpected cases in this topology too; Node 10 received more packets than Node 0, Node 1, Node 2 and Node 9, but when Node 10 changed into the attacker node, there were no packets lost during the simulation period.

As previously discussed, the study looked at 12 nodes with one attacker in the topology. According to the result of the experiment, it showed that the Nodes 1, 2 and 9 were more critical than other nodes because when they turned into the attack nodes, dropped more packets during the transfer process. The basic topology is still like the graph below with 12 nodes connected with each other in different ways, some of them connect with multiple neighbour nodes and some of them do not.

In this study, we aimed to discover how the network performed when there were two attackers in the network, so two nodes were turned into the attacker nodes while the others stayed the same. In general, there were 12 nodes which two out of 12 became attack nodes and one out of 12 was the final node. We changed the two attacker nodes from time to time and constructed a chart which shows all the dropping results.

Table 4.1 Dropping value of randomly changing two nodes into attackers in Case Study I

0&1	0	1&2	0	2&3	5	3&4	1	4&5	0
0&2	61	1&3	6	2&4	32	3&5	0	4&6	1
0&3	0	1&4	22	2&5	12	3&6	18	4&7	10
0&4	6	1&5	15	2&6	17	3&7	24	4&8	0
0&5	17	1&6	11	2&7	65	3&8	0	4&9	1
0&6	1	1&7	17	2&8	20	3&9	34	4&10	25
0&7	0	1&8	19	2&9	31	3&10	0		
0&8	0	1&9	30	2&10	21				
0&9	41	1&10	25						
0&10	8								

5&6	20	6&7	0	7&8	0	8&9	4	9&10	43
5&7	0	6&8	0	7&9	28	8&10	21		
5&8	0	6&9	39	7&10	0				
5&9	21	6&10	0						
5&10	0								

In this chart, we can get an initial picture about the dropping results and it shows that when Node 1, Node 2 and Node 9 were one of the attacker nodes, there were more packets discarded. The effect of Node 9 can not be seen clearly in this chart. The following line chart sums up the entire dropping packet picture, with the same nodes combined with the other nodes which performed as the attackers.

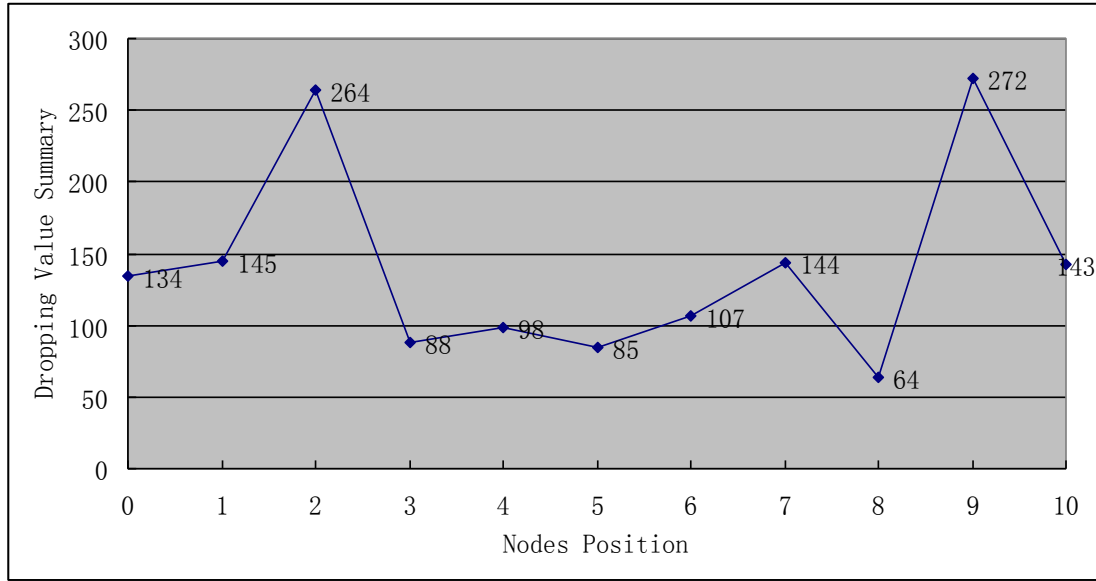


Figure 4.3- Dropping value summary for each node of Case Study I

According to the line chart above, it is clear that when Node 2 and Node 9 are one of the attackers, they can discard more packets during the whole packet passing process with 264 and 272 passes respectively. They are followed by Node 0, Node 1, Node 7 and Node 10 with 134, 145, 144 and 143 passes respectively. Node 3, Node 4, Node 5 and Node 6 had the fewer passes with 88, 98, 85 and 107 respectively. Lastly, when Node 8 was one of the attackers, it had the worst dropping behaviour with only 64 values in total.

Compared with the former case study, this time Node 7 and Node 10 had more impact on the final output performance, instead of there is no dropping at all when they act as single attack in the network. Besides that, it is clear that when one more attacker was added into the network, the dropping ratio rose, and more importantly, when Node 2 became one of the attackers, it almost caught up with the performance of Node 9, which outperformed the other nodes in the first case study.

Then we imported the topology into Gephi which can perform in-depth analysis of parameters with mathematical formulas. The imported topology is shown below:

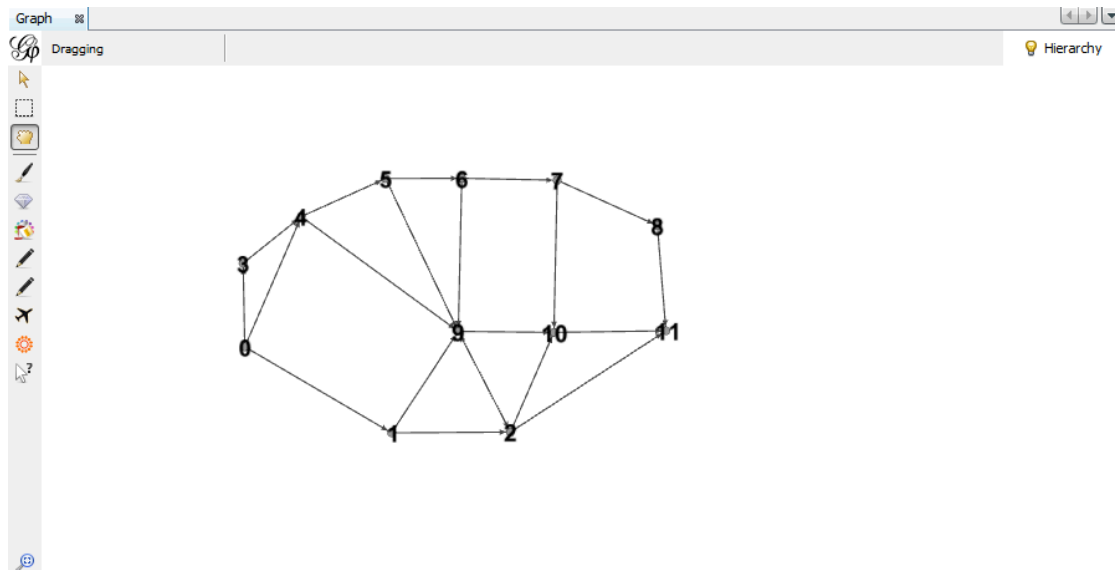


Figure 4.4- Case Study I Topology imported GUI in to Gephi

After importing the topology into the analysis software, we collected all the parameter evaluations (see Table 4.2) and the three distributions (betweenness centrality distribution, eccentricity distribution and closeness centrality distribution) and the results are shown below:

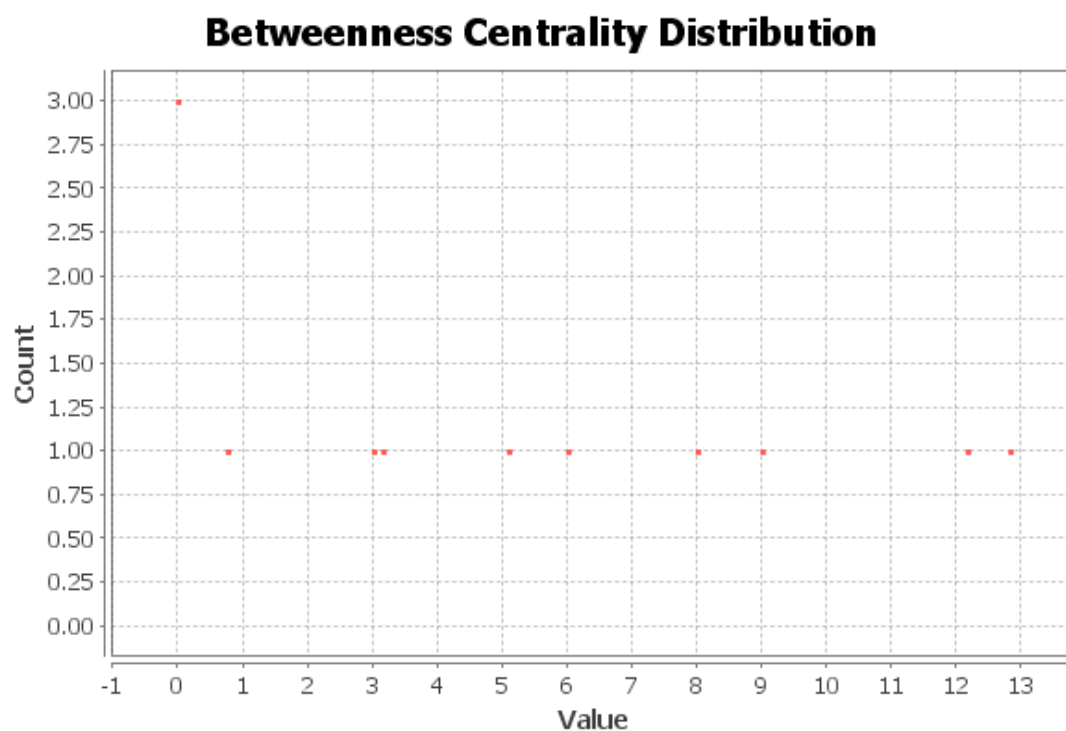


Figure 4.5 Betweenness centrality distribution of Case Study I

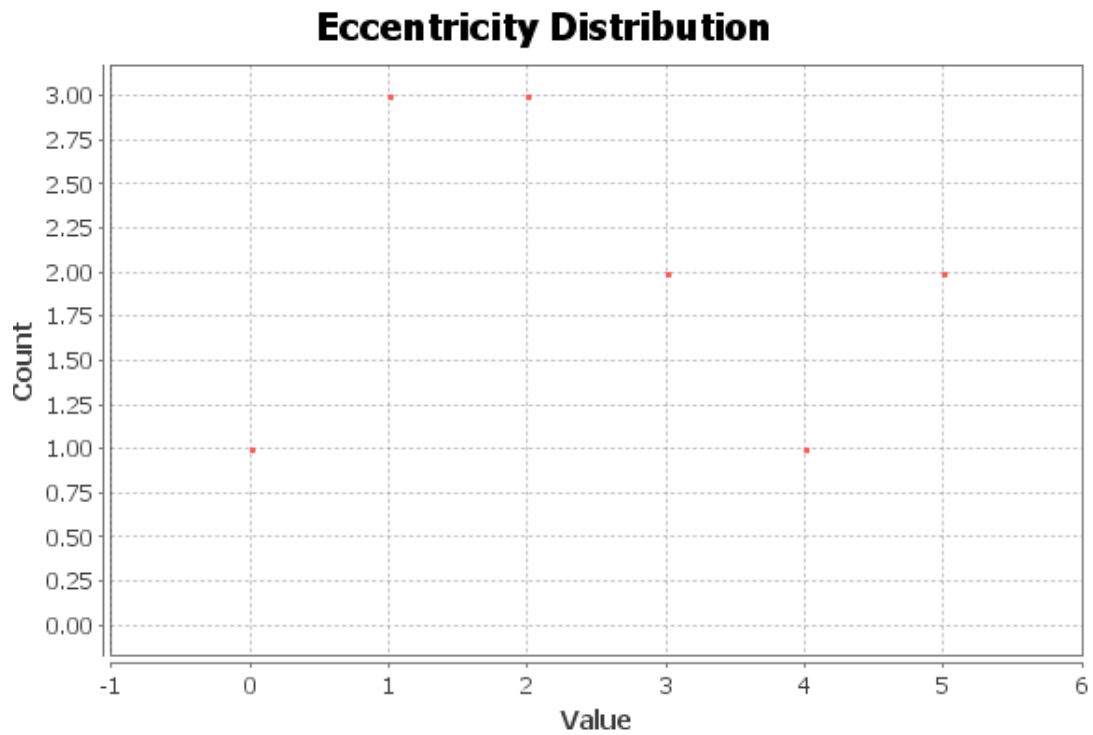


Figure 4.6 Eccentricity distribution of Case Study I

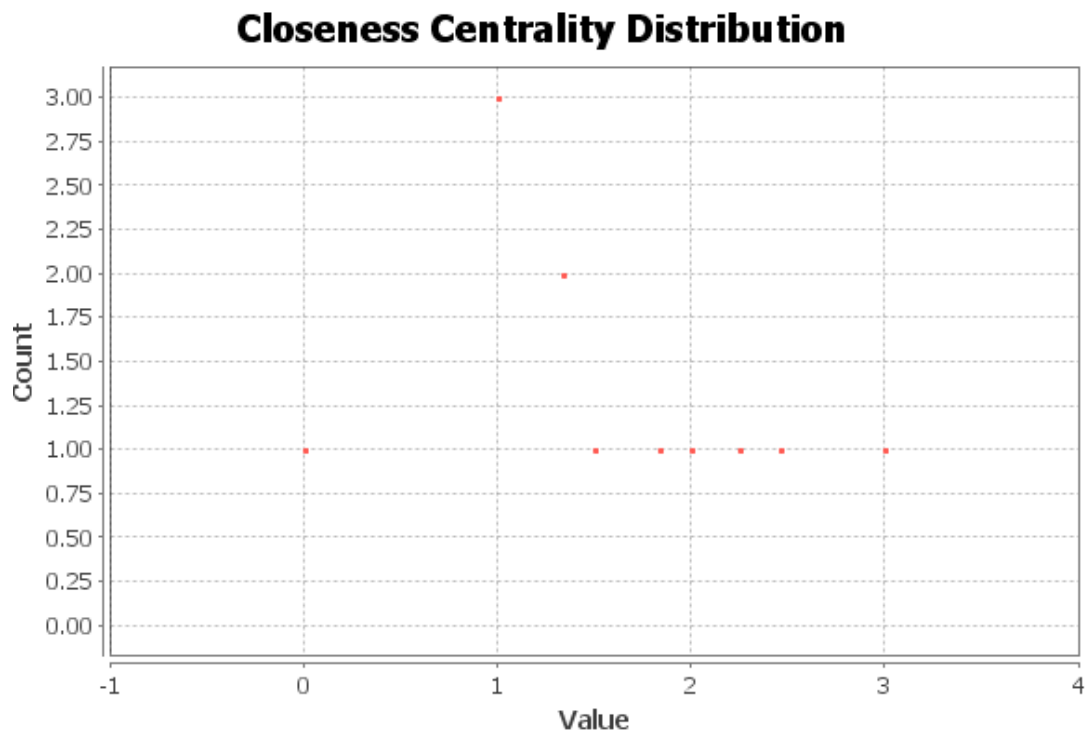


Figure 4.7 Closeness centrality distribution of Case Study I

In these three distribution graphs, we find that in betweenness centrality distribution, Nodes 1, 2 and 9 had higher values compared with the other nodes with 9.833, 8.5 and

20.337 respectively. According to the first experiment, Nodes 1, 2, and 9 also had a larger number of packets lost during the attack simulation process with values of 9, 9 and 18 respectively. So, betweenness is an effective metric to measure the attack nodes in this case.

Table 4.2 Metrics results of Case Study I topology

Id	Label	Eccent...	Closeness...	Betweenne...	Compo...	Strong...	In-Deg...	Out-D...	Degree	Weigh...	Weigh...	Weigh...	Authority	Hub	Modul...	PageRank	Clustering...	Eigenvert...
Node0	0	3	2.2	2.333	0	1	3	1	4	4	3	1	0.133	0.125	1	0.089	0.167	0.279
Node1	1	2	1.4	9.833	0	1	2	3	5	5	2	3	0.1	0.125	1	0.142	0.333	0.519
Node2	2	3	1.6	8.5	0	1	3	3	6	6	3	3	0.133	0.167	0	0.154	0.333	1
Node3	3	4	2.286	0	0	3	0	2	2	2	0	2	0	0.042	1	0.024	0.5	0
Node4	4	3	1.833	3.167	0	2	1	2	3	3	1	2	0.067	0.042	1	0.034	0.167	0.005
Node5	5	3	2.167	0	0	4	0	1	1	1	1	0	1	0.042	2	0.024	0	0
Node6	6	3	2	0	0	7	0	2	2	2	0	2	0	0.042	3	0.024	0	0
Node7	7	2	1.5	1.333	0	6	1	1	2	2	1	1	0.067	0.042	3	0.034	0	0.005
Node8	8	1	1	1.333	0	5	1	1	2	2	1	1	0.067	0.083	3	0.053	0	0.014
Node9	9	2	1.4	20.333	0	1	5	3	8	8	5	3	0.2	0.167	2	0.153	0.1	0.745
Node10	10	4	2.2	2.167	0	1	2	2	4	4	2	2	0.1	0.125	0	0.111	0.5	0.825
Node11	11	0	0	0	0	0	3	0	3	3	3	0	0.133	0	0	0.159	0.333	0.859

From the example above, we can infer the location of some critical nodes is the key to the whole topology structure. We can deduce some obvious differences from the parameters of these nodes and better understand why some nodes are so important.

4.2 Case study II

The second case study focused on testing the metric performance of each node and making a comparison with their dropping status. Moreover, there are three topologies in this case study which have slight differences among them. First of all, they all had 29 nodes and the first two topologies shared the same amount of links too. The differences between them are the first one was relatively narrow in overall structure, and the second was squarer. However, the third figure compared to the previous two had a smaller amount of links, but it covered more dimensions than the other two.

4.2.1 Three Topologies

From the topology structure below, it can be seen that three of the topologies consisted of 29 nodes connected by 68, 68, and 57 links accordingly. The topology structures are shown below:

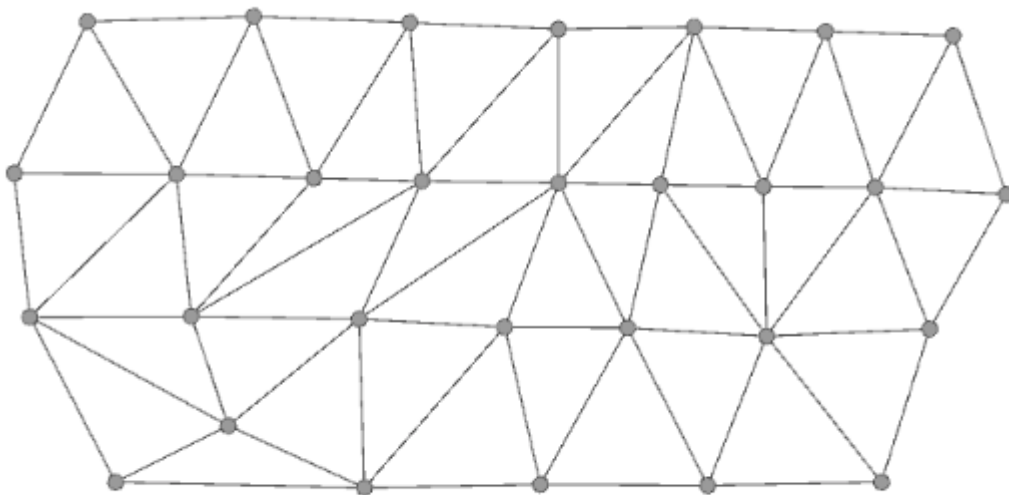


Figure 4.8 Structure of topology A: 29 Nodes and 68 links

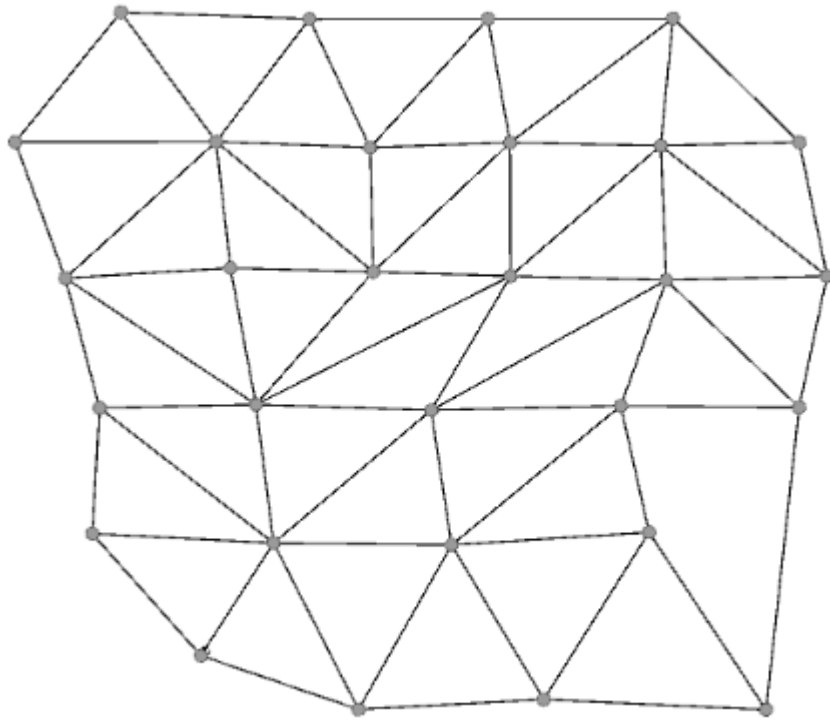


Figure 4.9 Structure of topology B: 29 Nodes and 68 links

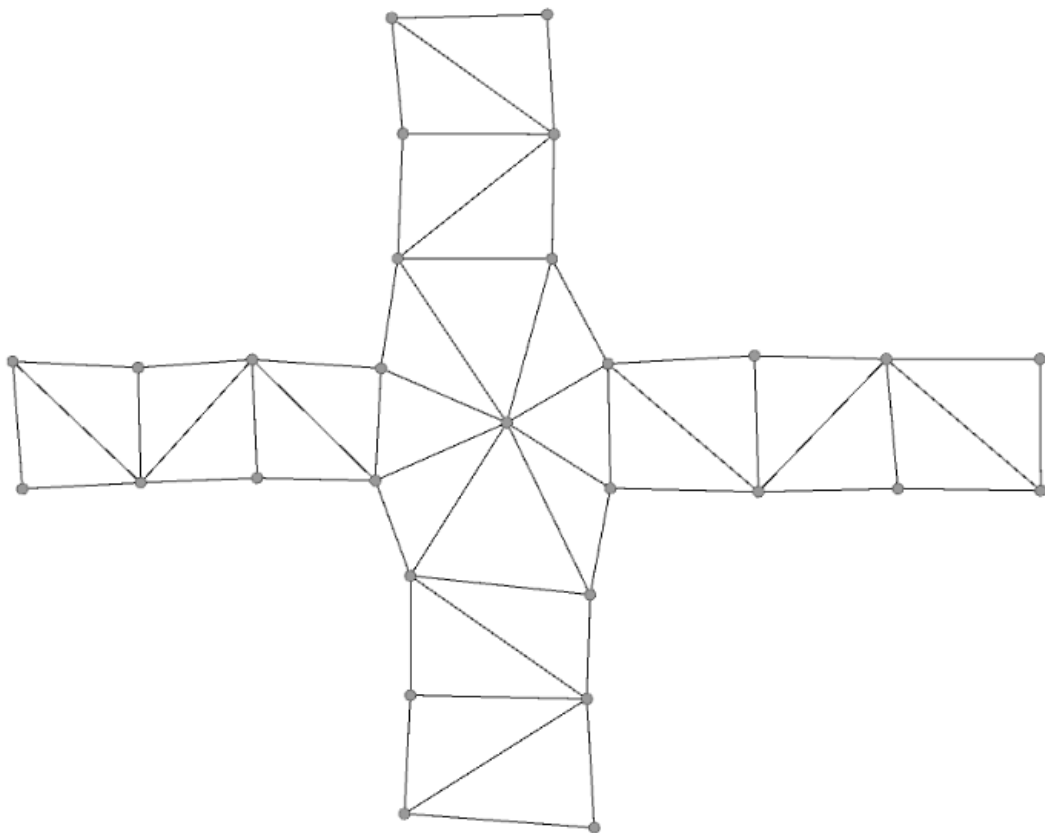


Figure 4.10 Structure of topology C: 29 Nodes and 57 links

4.2.2 Result Analysis

Next, we import these three topologies into Gephi for parameter evaluation and then put them into OMNeT++ for attack simulation, and the results shows below:

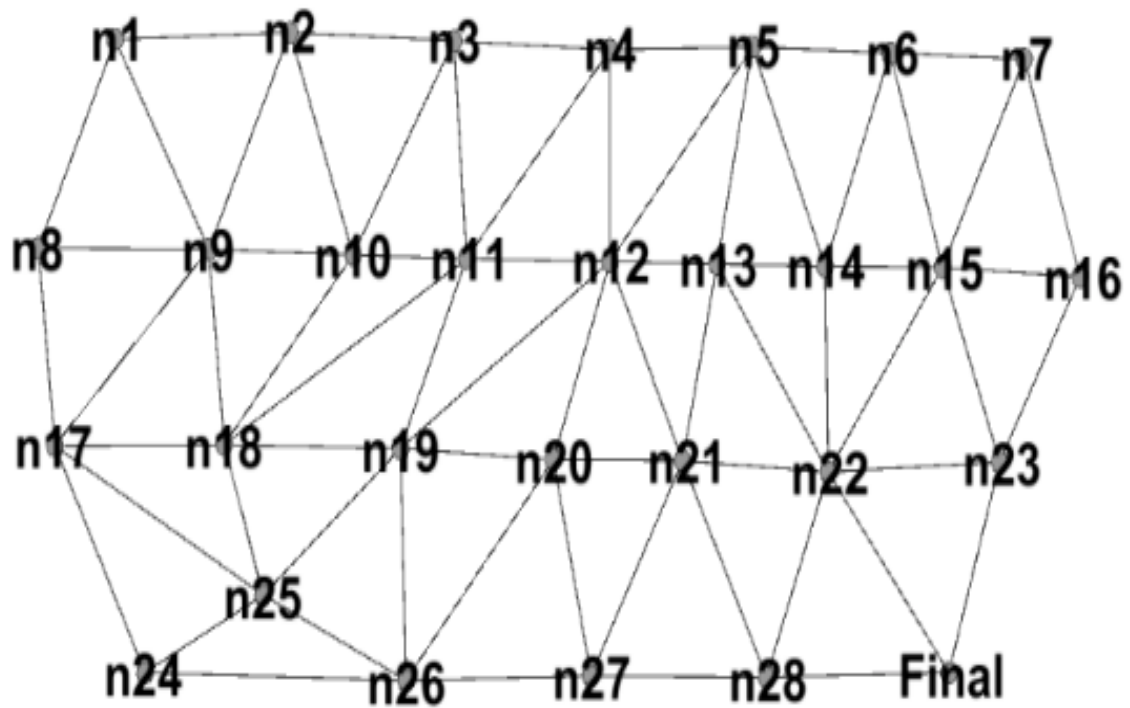


Figure 4.11 Topology A with nodes label

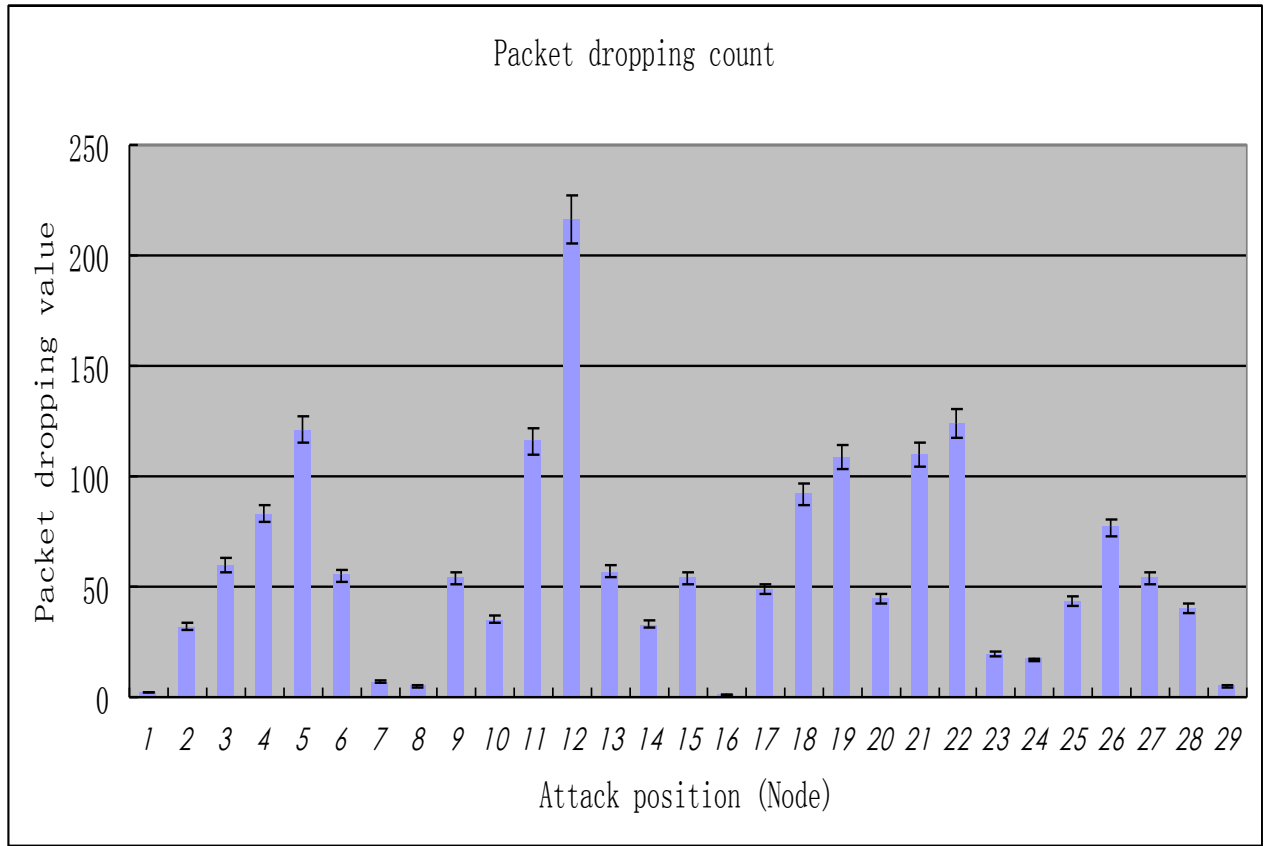


Figure 4.12 Topology A Packet dropping count

In topology A, 29 nodes had been attacked each time in OMNeT++ with a dropping attack during the simulation process. As the graph shows above, there were a lot of different values on each node, and each value bar had slightly value float too. We selected nodes 5, 11, 12, 19, 21, 22 as a sample. These six nodes can be seen as having the most dropping value with more than 100, and there were more than 200 values for Node 12, which was the highest. After that, we put topology A into the Gephi for nodes evaluation, and the results are shown in the next few pages. According to the table on the next page, we noticed that Nodes 5, 11, 12, 19, 21 and 22 have greater value of betweenness centrality distribution with 63.006, 57.579, 100.203, 53.095, 53.237, and 61.113, respectively.

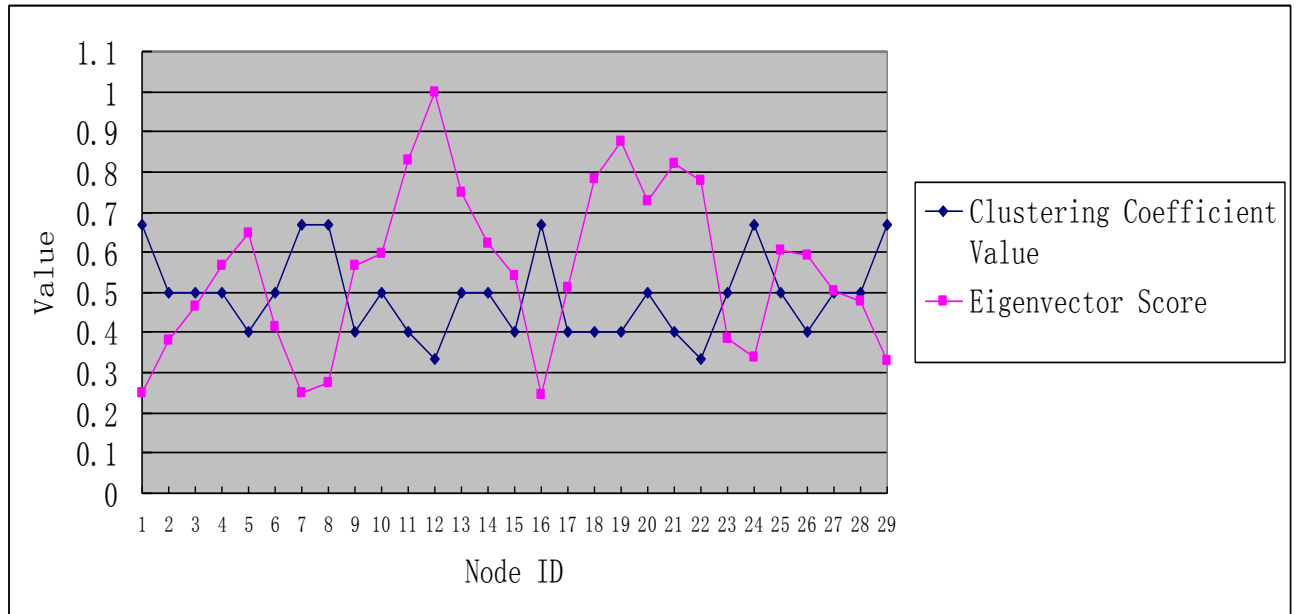


Figure 4.13 Clustering coefficient and Eigenvector performance of Topology A

Beside the betweenness performance of each node in the topology A, this figure shows the clustering coefficient of each node and their eigenvector score as well. According to the definition of clustering coefficient, it measures how nodes are embedded in their neighbourhood. As Figure 4.1.3 shows, Node 12 and Node 22 had the lowest clustering coefficient value which almost reached 0.3. Following that were Node 5, Node 9, Node 11, Node 15, Node 18, Node 19 and Node 21 which stayed at around a 0.4 value.

The eigenvector provided the analysis of influence of each node in the whole network which directly shows the importance of each node in the topology. From this figure, Node 12 scored 1 which means it is the most important node, while Node 22 is nearly 0.9, in second place. As a converse layout in this figure, the node which has the lowest clustering coefficient value has a more influential position in the topology. In topology A, Nodes 5, 11, 12, 19, 21 and 22 had more influence than other nodes and can cause more damage to the whole network when they turn into compromised nodes.

Table 4.3 Metric results of first fifteen nodes of topology A

Id	Label	Degree	Weighte...	Eccentricity	Closeness C...	Betweenness ...	Authority	Hub	Modularit...	PageRank	Compone...	Clustering C...	Number ...	Eigenvector...
Node1	n1	3	3	7	4	1.38	0.024	0.024	0	0.025	0	0.667	2	0.251
Node2	n2	4	4	6	3.429	15.79	0.03	0.03	0	0.031	0	0.5	3	0.379
Node3	n3	4	4	5	2.929	28.958	0.03	0.03	0	0.03	0	0.5	3	0.464
Node4	n4	4	4	4	2.571	40.985	0.03	0.03	1	0.029	0	0.5	3	0.566
Node5	n5	5	5	5	2.571	63.006	0.036	0.036	1	0.036	0	0.4	4	0.649
Node6	n6	4	4	6	3.107	27.419	0.03	0.03	2	0.031	0	0.5	3	0.415
Node7	n7	3	3	7	3.75	3.402	0.024	0.024	2	0.025	0	0.667	2	0.25
Node8	n8	3	3	8	4.071	2.164	0.024	0.024	0	0.025	0	0.667	2	0.273
Node9	n9	6	6	7	3.393	26.238	0.042	0.042	0	0.044	0	0.4	6	0.565
Node10	n10	5	5	6	2.964	17.583	0.036	0.036	0	0.036	0	0.5	5	0.597
Node11	n11	6	6	5	2.464	57.579	0.042	0.042	1	0.041	0	0.4	6	0.828
Node12	n12	7	7	4	2.25	100.203	0.048	0.048	1	0.047	0	0.333	7	1
Node13	n13	5	5	5	2.571	26.738	0.036	0.036	1	0.035	0	0.5	5	0.749
Node14	n14	5	5	6	2.929	17.073	0.036	0.036	2	0.036	0	0.5	5	0.621
Node15	n15	6	6	7	3.357	26.808	0.042	0.042	2	0.045	0	0.4	6	0.548

Table 4.4 Metric results of rest fourteen nodes of topology A

Node16	n16	3	3	8	4.214	0.833	0.024	0.024	2	0.026	0	0.667	2	0.247
Node17	n17	5	5	7	3.321	23.753	0.036	0.036	3	0.037	0	0.4	4	0.513
Node18	n18	6	6	6	2.786	44.454	0.042	0.042	3	0.042	0	0.4	6	0.782
Node19	n19	6	6	5	2.429	53.095	0.042	0.042	3	0.041	0	0.4	6	0.874
Node20	n20	5	5	4	2.5	22.751	0.036	0.036	4	0.035	0	0.5	5	0.727
Node21	n21	6	6	5	2.5	53.237	0.042	0.042	4	0.041	0	0.4	6	0.819
Node22	n22	7	7	6	2.857	61.113	0.048	0.048	2	0.05	0	0.333	7	0.778
Node23	n23	4	4	7	3.607	9.075	0.03	0.03	2	0.032	0	0.5	3	0.387
Node24	n24	3	3	6	3.286	7.658	0.024	0.024	3	0.024	0	0.667	2	0.338
Node25	n25	5	5	6	2.857	20.238	0.036	0.036	3	0.036	0	0.5	5	0.606
Node26	n26	5	5	5	2.786	36.969	0.036	0.036	3	0.036	0	0.4	4	0.591
Node27	n27	4	4	5	2.857	25.979	0.03	0.03	4	0.029	0	0.5	3	0.505
Node28	n28	4	4	6	2.929	19.885	0.03	0.03	4	0.03	0	0.5	3	0.476
Node29	Final	3	3	7	3.5	2.634	0.024	0.024	2	0.024	0	0.667	2	0.328

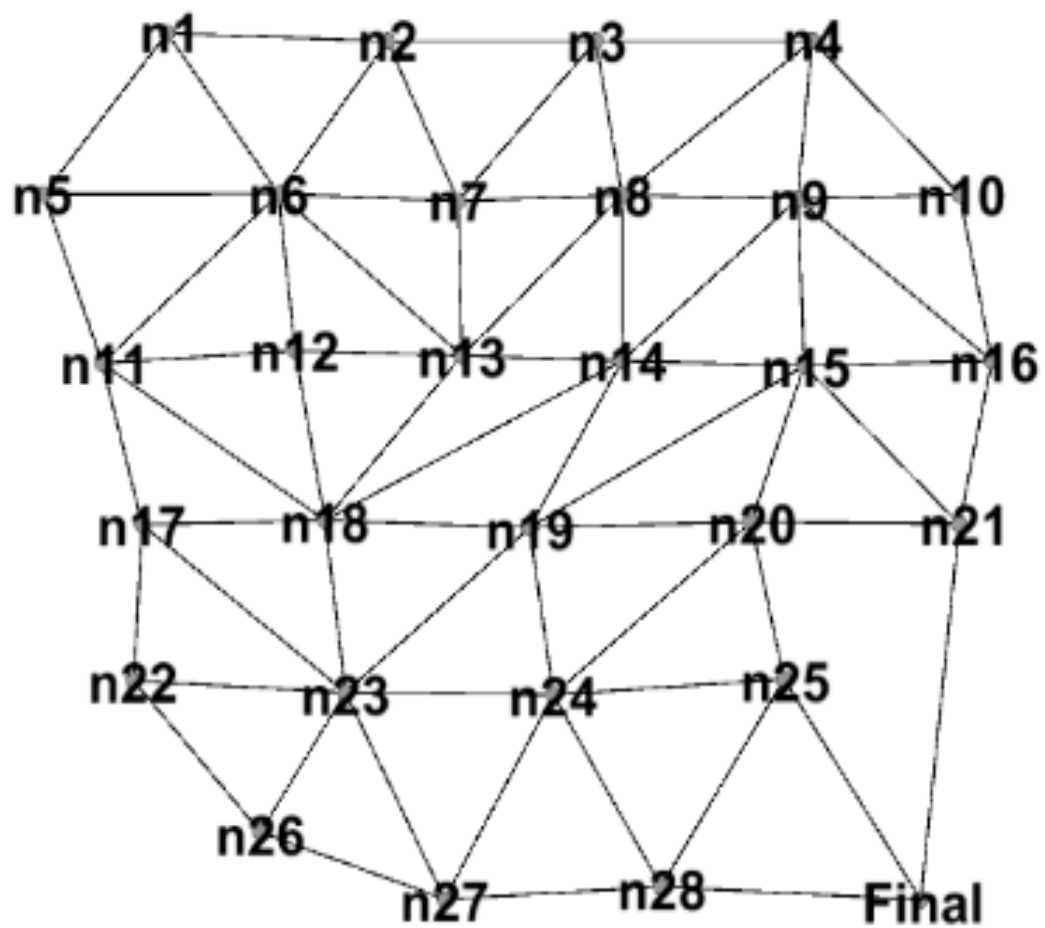


Figure 4.14 Topology B with nodes label

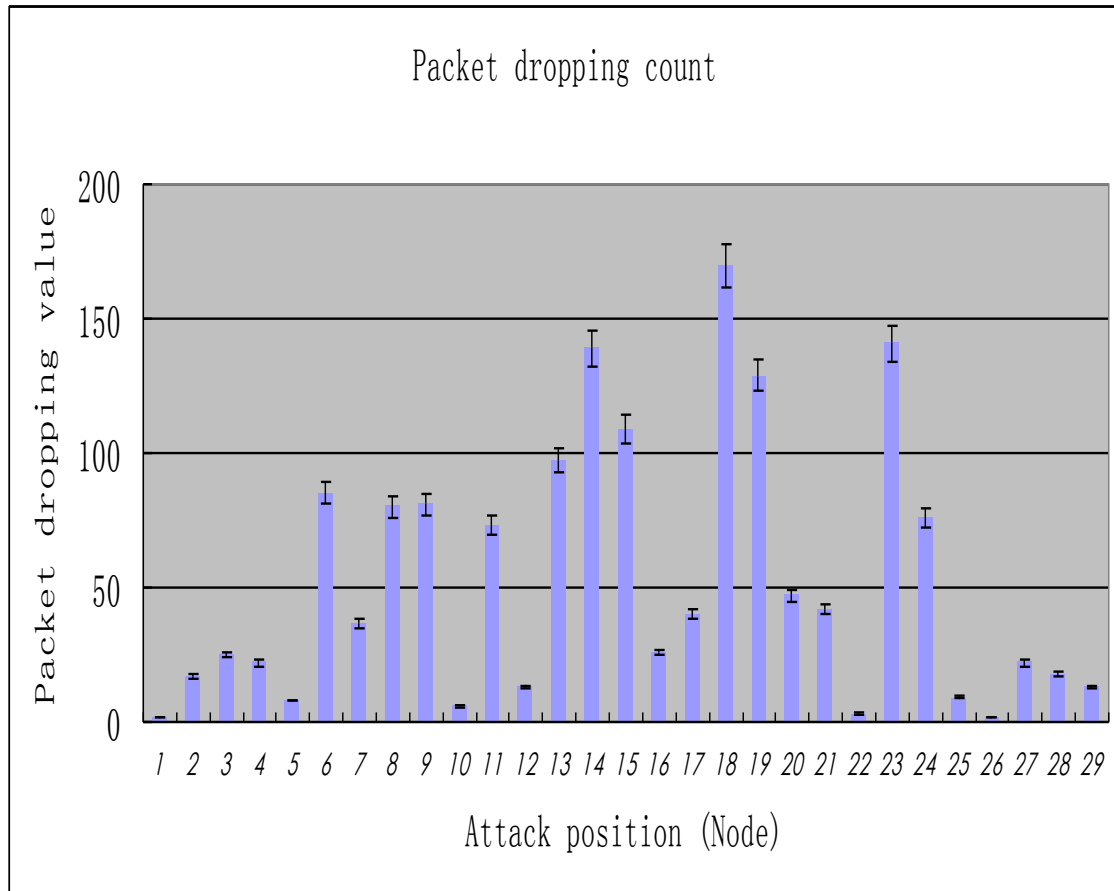


Figure 4.15 Topology B Packets dropping count

In topology B, 29 nodes had been attacked each time in the OMNeT++ with a dropping attack during the simulation process. As the graph above shows, there were a lot of different values on each node, and each value bar had a slightly float too. We selected Nodes 14, 15, 18, 19, 23 as an example. These five nodes can be seen as having the most dropping value from 100 to 250 and there node 18 had a value over 200 , which was the highest. After that, we put topology B into the Gephi for nodes evaluation, and the results are in Table 4.5 and 4.6. According to the table, Nodes 14, 15, 18, 19, 23 had greater values of betweenness centrality distribution with 68.147, 50.555, 84.707, 63.881, and 69.267 respectively.

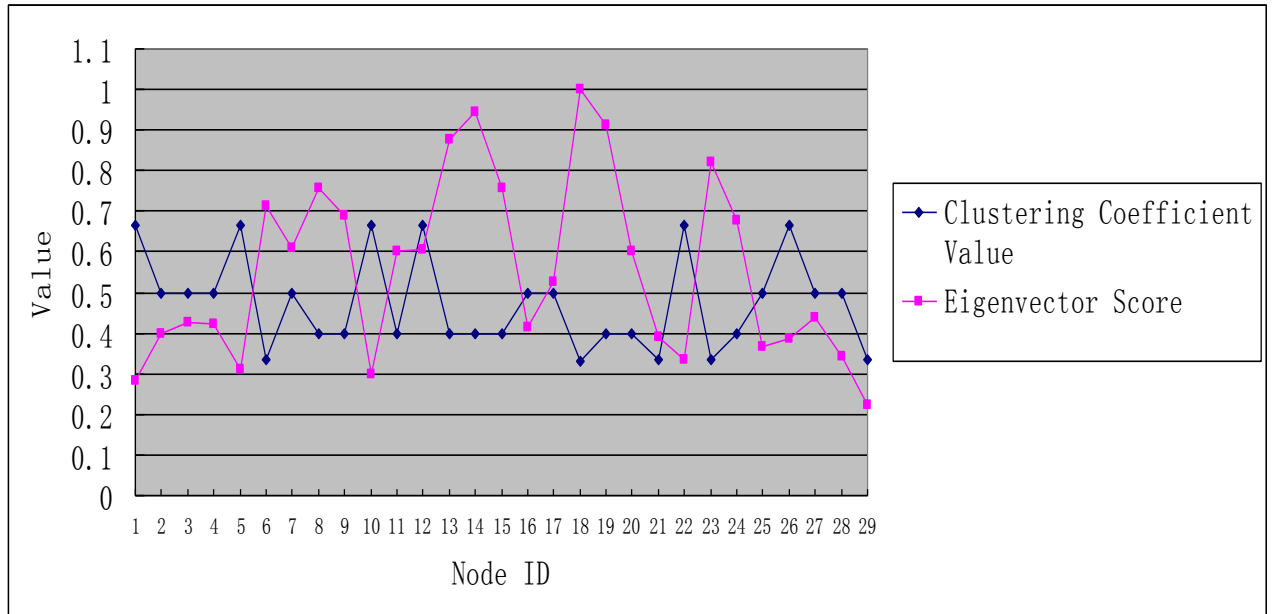


Figure 4.16 Clustering coefficient and Eigenvector performance of Topology B

This figure shows the clustering coefficient of each node and their eigenvector score in topology B. As Figure 4.16 shows, Nodes 6, 18, 21, and 23 had the lowest clustering coefficient value which almost reached 0.3. Following that were Nodes 8, 9, 11, 13, 14, 15, 19, 20, and 24 which stand around a 0.4 value.

From this figure, Node 18 scores 1 which means it is the most important node, while Node 14 was second place at nearly 0.95. So, in the case of topology B, Nodes 14 and 18 had more influence than the other nodes so these two nodes can cause more damage to the whole network when they turn into compromised nodes.

Table 4.5 Metric results of first fifteen nodes of topology B

Id	Label	Degree	Waighte...	Eccentricity	Closeness C...	Betweenness ...	Authority	Hub	Modularit...	PageRank	Compone...	Clustering C...	Number o...	Eigenvector ...	
Node1	n1	3	3	3	6	3.571	1.144	0.024	0.024	0	0.024	0	0.667	2	0.281
Node2	n2	4	4	4	6	3.357	8.368	0.03	0.03	0	0.03	0	0.5	3	0.398
Node3	n3	4	4	4	5	3.107	12.15	0.03	0.03	1	0.03	0	0.5	3	0.425
Node4	n4	4	4	4	5	3.036	10.868	0.03	0.03	1	0.03	0	0.5	3	0.421
Node5	n5	3	3	3	6	3.286	3.96	0.024	0.024	0	0.024	0	0.667	2	0.311
Node6	n6	7	7	7	5	2.786	41.292	0.048	0.048	0	0.049	0	0.333	7	0.712
Node7	n7	5	5	5	5	2.786	15.045	0.036	0.036	0	0.035	0	0.5	5	0.611
Node8	n8	6	6	6	4	2.5	39.426	0.042	0.042	1	0.041	0	0.4	6	0.759
Node9	n9	6	6	6	4	2.536	39.369	0.042	0.042	1	0.042	0	0.4	6	0.69
Node10	n10	3	3	3	5	3.214	2.515	0.024	0.024	1	0.024	0	0.667	2	0.298
Node11	n11	5	5	5	5	2.643	34.215	0.036	0.036	0	0.035	0	0.4	4	0.602
Node12	n12	4	4	4	5	2.643	5.918	0.03	0.03	0	0.028	0	0.667	4	0.606
Node13	n13	6	6	6	4	2.286	48.907	0.042	0.042	0	0.04	0	0.4	6	0.878
Node14	n14	6	6	6	3	2.107	68.147	0.042	0.042	1	0.039	0	0.4	6	0.945

Table 4.6 Metric results of rest fourteen nodes of topology B

Node15	n15	6	6	4	2.357	50.555	0.042	0.042	3	0.041	0	0.4	6	0.756
Node16	n16	4	4	5	2.929	12.548	0.03	0.03	1	0.03	0	0.5	3	0.415
Node17	n17	4	4	4	2.607	19.113	0.03	0.03	2	0.028	0	0.5	3	0.526
Node18	n18	7	7	4	2.143	84.707	0.048	0.048	0	0.045	0	0.333	7	1
Node19	n19	6	6	4	2.179	63.881	0.042	0.042	3	0.039	0	0.4	6	0.913
Node20	n20	5	5	5	2.643	22.522	0.036	0.036	3	0.035	0	0.4	4	0.6
Node21	n21	4	4	5	2.964	21.231	0.03	0.03	3	0.03	0	0.333	2	0.39
Node22	n22	3	3	5	3.107	1.783	0.024	0.024	2	0.021	0	0.667	2	0.336
Node23	n23	7	7	4	2.393	69.267	0.048	0.048	2	0.045	0	0.333	7	0.821
Node24	n24	6	6	5	2.643	37.131	0.042	0.042	3	0.041	0	0.4	6	0.676
Node25	n25	4	4	6	3.25	4.633	0.03	0.03	3	0.03	0	0.5	3	0.366
Node26	n26	5	4	5	3.214	1.083	0.03	0.03	2	0.027	0	0.667	4	0.388
Node27	n27	4	4	5	3.036	10.146	0.03	0.03	2	0.028	0	0.5	3	0.437
Node28	n28	4	4	6	3.321	8.042	0.03	0.03	3	0.03	0	0.5	3	0.341
Node29	Final	3	3	6	3.5	6.033	0.024	0.024	3	0.025	0	0.333	1	0.222

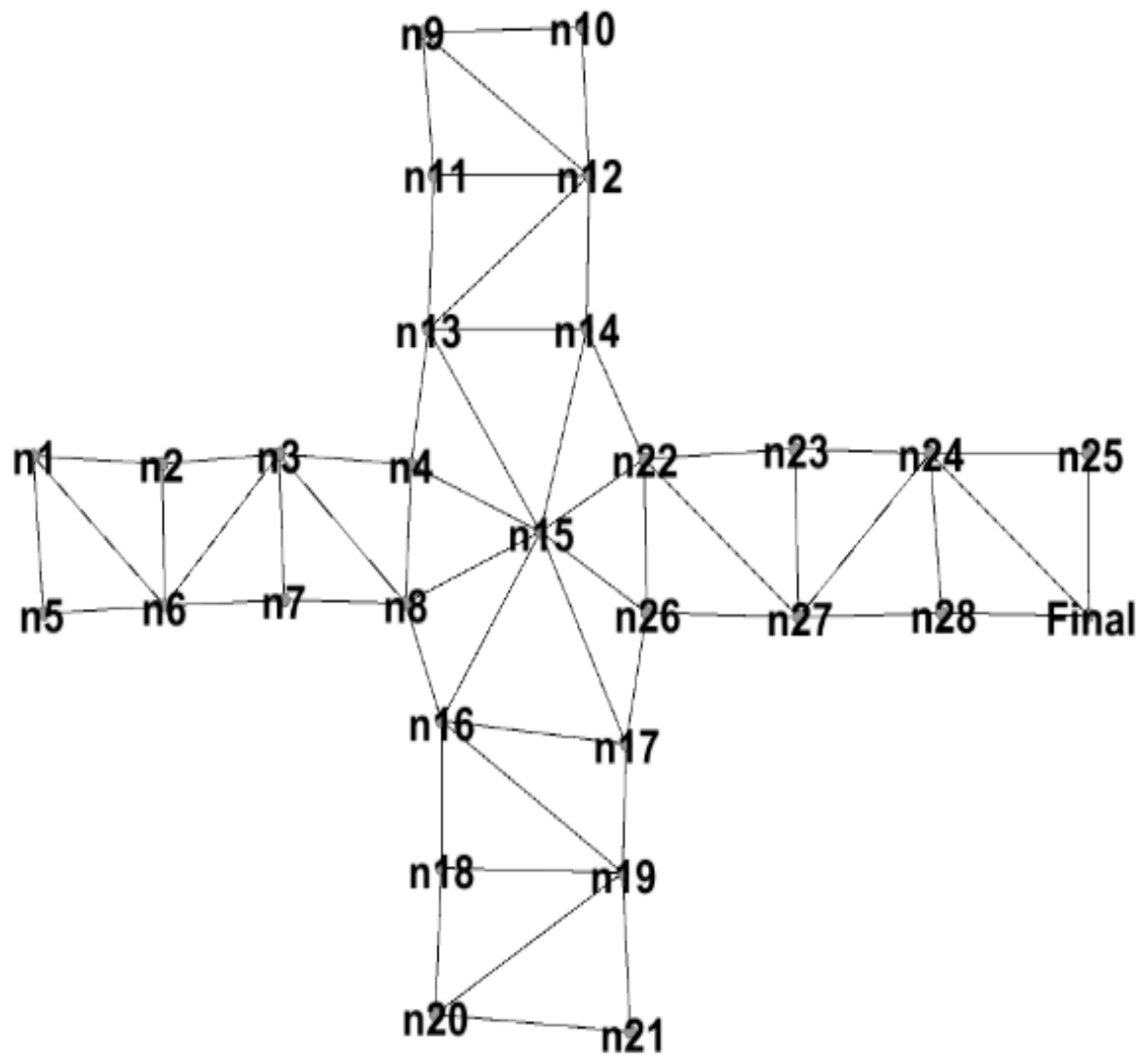


Figure 4.17 Topology C with nodes label

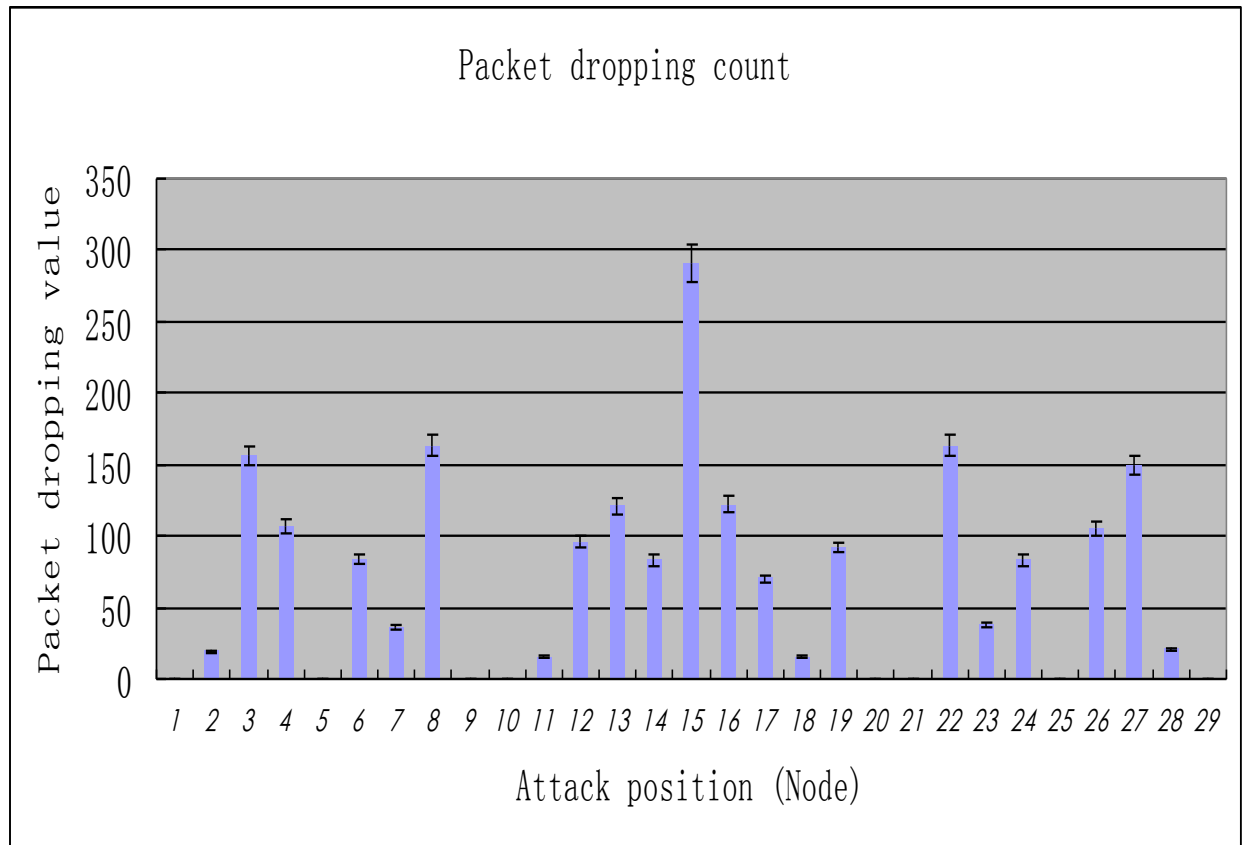


Figure 4.18 Topology C Packet dropping count

In topology C, 29 nodes had been attacked each time in the OMNeT++ with a dropping attack during the simulation process. As the graph above shows, there were a lot of different values on each node, and each value bar had a slight value float too. We selected Nodes 4, 8, 13, 15, 16, 22, 26, and 27 as examples. These eight nodes had the highest dropping value from 100 to 300 this time and there were more than 300 values for Node 15, which was the highest. After that, we put topology C into the Gephi for nodes evaluation. According to the table on the next page, we noticed that Nodes 4, 8, 13, 15, 16, 22, 26, and 27 had greater values of betweenness centrality distribution too with 52.793, 79.293, 58.765, 146.541, 61.755, 79.728, 50.639, and 76.233 respectively.

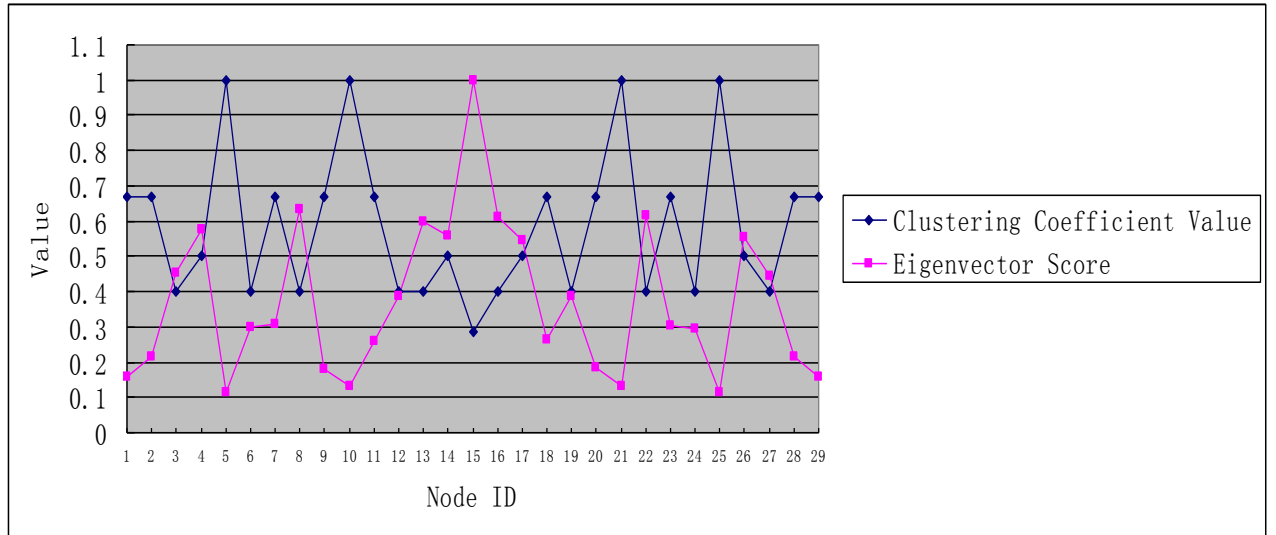


Figure 4.19 Clustering coefficient and Eigenvector performance of Topology C

This figure shows the clustering coefficient of each node and their eigenvector score in topology C. As Figure 4.19 shows, Node 15 had the lowest clustering coefficient value, which was below 0.3. Following that were Nodes 3, 6, 8, 13, 16, 19, 22, 24, and 27, which had a value around 0.4.

From this figure, Node 15 scored 1, which made it the most important node. So, in the case of topology C, Node 15 had the most influence which made it the most critical node in the whole network when it turned into a compromised node.

Table 4.7 Metric results of first fifteen nodes of topology C

Id	Label	Degree	Weighte...	Eccentricity	Closeness C...	Betweenness ...	Authority	Hub	Modularit...	PageRank	Compone...	Clustering C...	Number ...	Eigenvector ...
Node1	n1	3	3	8	4.679	0.5	0.028	0.028	0	0.03	0	0.667	2	0.158
Node2	n2	3	3	7	3.857	9.904	0.028	0.028	0	0.029	0	0.667	2	0.217
Node3	n3	5	5	6	3.036	76.96	0.043	0.043	0	0.043	0	0.4	4	0.454
Node4	n4	4	4	5	2.643	52.793	0.035	0.035	0	0.033	0	0.5	3	0.575
Node5	n5	2	2	8	4.714	0	0.021	0.021	0	0.022	0	1	1	0.115
Node6	n6	5	5	7	3.786	41.096	0.043	0.043	0	0.046	0	0.4	4	0.298
Node7	n7	3	3	6	3.286	17.493	0.028	0.028	0	0.027	0	0.667	2	0.31
Node8	n8	5	5	5	2.607	79.293	0.043	0.043	0	0.041	0	0.4	4	0.635
Node9	n9	3	3	6	4.107	0.5	0.028	0.028	1	0.03	0	0.667	2	0.182
Node10	n10	2	2	6	4.143	0	0.021	0.021	1	0.021	0	1	1	0.133
Node11	n11	3	3	6	3.536	6.912	0.028	0.028	1	0.028	0	0.667	2	0.261
Node12	n12	5	5	5	3.214	46.099	0.043	0.043	1	0.045	0	0.4	4	0.386
Node13	n13	5	5	5	2.714	58.765	0.043	0.043	1	0.041	0	0.4	4	0.598
Node14	n14	4	4	5	2.714	40.54	0.035	0.035	1	0.033	0	0.5	3	0.557

Table 4.8 Metric results of rest fourteen nodes of topology C

Node15	n15	8	8	4	2.286	146.541	0.064	0.064	1	0.061	0	0.286	8	1
Node16	n16	5	5	5	2.679	61.753	0.043	0.043	2	0.041	0	0.4	4	0.61
Node17	n17	4	4	5	2.75	36.997	0.035	0.035	2	0.033	0	0.5	3	0.547
Node18	n18	3	3	6	3.5	7.183	0.028	0.028	2	0.028	0	0.667	2	0.264
Node19	n19	5	5	5	3.214	45.272	0.043	0.043	2	0.045	0	0.4	4	0.387
Node20	n20	3	3	6	4.107	0.5	0.028	0.028	2	0.03	0	0.667	2	0.183
Node21	n21	2	2	6	4.143	0	0.021	0.021	2	0.021	0	1	1	0.133
Node22	n22	5	5	5	2.643	79.728	0.043	0.043	1	0.041	0	0.4	4	0.616
Node23	n23	3	3	6	3.321	17.801	0.028	0.028	3	0.027	0	0.667	2	0.305
Node24	n24	5	5	7	3.857	41.122	0.043	0.043	3	0.046	0	0.4	4	0.295
Node25	n25	2	2	8	4.786	0	0.021	0.021	3	0.022	0	1	1	0.114
Node26	n26	4	4	5	2.679	50.639	0.035	0.035	1	0.033	0	0.5	3	0.558
Node27	n27	5	5	6	3.107	76.233	0.043	0.043	3	0.043	0	0.4	4	0.445
Node28	n28	3	3	7	3.929	9.878	0.028	0.028	3	0.029	0	0.667	2	0.215
Node29	Final	3	3	8	4.75	0.5	0.028	0.028	3	0.03	0	0.667	2	0.157

Table 4.9 Five metric comparison of three topologies

Topology Name	Average Degree	Average Weight Degree	Average Clustering Coefficient	Average Path Length	Graph Density
Topology A	4.69	4.69	0.492	3.062	0.167
Topology B	4.759	4.724	0.475	2.833	0.17
Topology C	3.862	3.862	0.585	3.475	0.138

The table above shows five topology evaluation parameters which make a comparison of these three similar topologies. From this table, the degree metric indicated the number of links, known as edges in the structure, which topology A and B had nearly the same value of around 4.7, while topology C had a value of 3.862. According to the number of links of each topology at the very beginning, there were 68 links in both topology A and B, while there were 57 links in topology C.

The average clustering coefficient however shows how nodes are embedded in their neighbourhood. From this table, topology B had the better performance with a clustering coefficient of 0.475, followed by topology A with 0.492, and 0.585 for topology C. As well as in average path length, topology B still remained the smallest with 2.833, while topology A and C were 3.062 and 3.475 respectively. These two metrics depend on how close the nodes engaged with each other, which can generally be inferred from the shape of these three structures. Topology B is more centralized than topology A despite having the same number of links and nodes. Topology C's structure was located more widely with even fewer links.

4.3 Summary of case study:

In Case Study 1, a topology with 12 nodes had been simulated under a dropping attack, and after transforming each node into the attackers, the whole packet loss

value was summarized and listed in Figure 4.2. Figure 4.3 along with Table 4.1 indicated the total dropping performance when two random attackers were placed in the topology. According to the metric analysis result from Figure 4.5 and Table 4.2, it is clear that the nodes which have high betweenness centrality distribution drop more packets when they turn into the compromised nodes (attackers). To sum up the first case study, it can be said that the betweenness centrality distribution is one of the metrics that could show the importance of each node in the whole topology.

The second case study, however, showed more metric evaluations in each topology. Three topologies were introduced which had similar features. All three topologies had 29 nodes and Topology A and B had 68 links while Topology C had 57 links. The difference between Topology A and B is they had a slightly different structure arrangement; Topology A's layout was a rectangle and Topology B's was a circle. After that, we analysed the betweenness distribution, clustering coefficient, and eigenvector of each node in each topology, along with their packets dropping performance. The results showed that the nodes which had the highest packets dropping value all had higher betweenness distribution and lower clustering coefficient value and had a large score in the eigenvector. Last but not least, we used five metrics to compare these three topologies and the comparison results are located in Table 4.9. According to the performance of the average clustering coefficient, average path length and graph density, the overall performance of Topology B is better than the other two and it had less dropping value as well. However, the decent performance of Topology B in these areas does not make it stand out from some other metric measures such as average degree and average weight degree, as the statistical data of Topology A and B is almost equal with each other.

Chapter 5

Conclusions and future work

5.1 Conclusion

In this section, we summarized the main contribution of each chapter and give a final conclude about how metric have influence on the topologies. Besides that, some discussion about further study is mentioned as well.

In the second chapter, we indicated three major network attacks which we used for the simulation in this thesis. They are delay attack, dropping attack and sinkhole attack. By introducing them in the second chapter, we have a better understanding of the characteristics of the three attacks and each attack has different effects and purposes. With intuitive attacks, their main purpose is to interfere with the flow of data from the sender's side to most terminals through some delay or dropping function.

In the third chapter, we discussed the four different types of topology structures. They are linear, star, ring and mesh topology. These four topologies have their own advantages and disadvantages, and it is clear that these factors become the weak link for cyber attackers to target their attacks towards systems. Compared with these four topology structures, we are more likely familiar with the structure of the first three because they are more simple and intuitive. However, because the structure is too simple, that is, most of the nodes connected to each other with one single link, once an attack happens in a certain part of structure, the entire topology could be paralyzed. So compared to the previous three structures, the fourth mesh topology structure has more diversified links between the nodes and the connection between the nodes will

be more complicated, so when a node is attacked, there will be other paths to choose from.

Chapter 3 also describes the contents of a number of metrics and simulation parameters that is part of the experimental analysis section of this thesis. The purpose of this chapter was to objectively assess the relationship between the whole topology and each node or the adjacent node, including distance, number of links, the amount of coverage and so on. Through these acquisition parameters, we can be more intuitive about which node has a more important position; normally these nodes are often act as an important key for attack and defence.

Chapter 4 is the core of this thesis, as we compared a series of experiments and data in this chapter. The first example is our own design topology structure, which contained 12 nodes and 20 links. We launched the attack at each node in this topology. Attack type was implemented as a dropping attack. According to the final result, by comparing with each node, the Nodes 1, 2 and 9 had more missing packets during the attack process with 9, 9, and 18 packets respectively. Then, we attacked any two nodes in this topology structure at the same time. Through simulation and data summary, we constructed a table to summarize all the packet loss results. The two experiments showed similar results. We speculated whether these three nodes had some kind of significant different parameter values compared with others.

We imported this topology into analysis software for more in-depth evaluation. We used Gephi which can analyze mathematical formulas and categories for each node of this topology. We put the results of the evaluation in the fifth chapter. Through analysis, we found that the Nodes 1, 2, and 9 had a greater betweenness centrality distribution with 9.833, 8.5, and 20.333, respectively. In order to verify the reliability of the conclusions reached at the beginning of the case studies, we changed the order of the experiment. The difference was that we introduced another three topologies which had several similar features in their architecture. They all had 29 nodes and the

first two had the same amount of links too, but with a slightly different range of structural coverage. The third topology had 57 links, while the former two had 68.

First, we carried out these three topology structure analyses by evaluating each node. In the first topology map, Nodes 5, 11, 12, 19, 22 had a large betweenness centrality distribution with 63.006, 57.579, 100.203, 53.095, 53.237, and 61.113, respectively. In the second topology map, Nodes 14, 15, 18, 19, and 23 had larger measurement values with 68.147, 50.555, 84.707, 63.881, and 69.267 respectively. In the third topology map, the measures of the larger nodes were 4, 8, 13, 15, 16, 22, 26, and 27. Measurement values were 52.793, 79.293, 58.765, 146.541, 61.755, 79.728, 50.639, and 76.233 respectively.

Interestingly, in the third topology, it can be observed that Nodes 5, 10, 21, and 25 have no betweenness centrality distribution which has the value of 0. In the third topology, the position of these four nodes is all located in the corner and have only two links. The most linked node in Topology C is Node 15 which owns eight links at the same time. From the betweenness distribution results, it has the highest value in all three topologies with 146.541 and its dropping amount is the largest with 291 values. Then we put these three topologies into OMNeT++ simulation software for an attack simulation. The results matched our previous conclusions, that the betweenness centrality distribution determines the flow of data throughput within connected nodes, and that means the higher betweenness centrality distribution leads to nodes becoming critical. And because of that, those critical nodes are more likely to become the targets of attack, so they also need more protection.

Through these experiments, it is not difficult to find these important nodes have a substantially greater amount of links. The majority of their positions is at the centre of the structure. And by studying the three topologies, we can find the structure and the amount of links play a vital role in the topology, although the first two share the same nodes and links, there are fewer packets lost in the second topology. By comparing

these two structures, we can find that the second topology has more square or round coverage area. The first one is more elongated. And a third one is like a cross topology and has the fewest links on four small branches. The results show that the nodes which have a high packets dropping value all have higher betweenness distribution and lower clustering coefficient values and have a large score in the eigenvector. Last but not least, we used five metrics to compare these three topologies and the comparison results are located in Table 4.9. According to the performance of the average clustering coefficient, average path length and graph density, the overall performance of Topology B is better than the other two and it has a lower dropping value as well. However, the decent performance of Topology B can not be concluded clearly from some other metrics such as average degree and average weight degree as the statistical data of Topology A and B are almost equal with each other.

By this phenomenon, we can preliminary conclude that when two topologies have the same amount of links, the more centralized structure topology is more secure. And the metrics such as betweenness centrality, clustering coefficient and eigenvector can more effectively show which nodes are more critical in the network. And the metric of average clustering coefficient, average path length and graph density can give convincing proof of the better topology while metrics like average degree and average weight degree can not.

5.2 Future work

Throughout this research, we have collected a large number of topology measurement data, and although this thesis involved the conduction of many experiments, we still need more in-depth discussion in many fields. Those possibilities for further studies are listed below:

- 1: We need to simulate more different types of attacks on the same topology in order to have a better evaluation of defence ability. In this thesis, the dropping attack is

chosen in the simulation process. However, due to the different characteristics and attack purposes, the metrics evaluation results might be different too.

2: We need to simulate more realistic attack scenarios, such as adjusting the topology architecture during the attack or launch more than one attack in the same topology at the same time. In the simulation part of this thesis, one single attack was launched during the whole simulation process. However, the situation might be more complex in the reality which has several attacks at the same time. As previews review, the wormhole attack, as an example, usually works with other attacks such as blackhole attack [91, 93].

3: We need to design more defensive means to protect topology; after all, the ultimate goal of the study of is to protect wireless networks from network attacks. This thesis focuses on the evaluation about how attacks affect the topology. By analysis the metrics results, some metrics can be defined have much affection on the attacks. However, there is no further discussion about how to take protective measures on the topology.

References

- [1] Akyildiz, I. F., & Wang, X. (2005). A survey on wireless mesh networks. *Communications Magazine, IEEE*, 43(9), S23-S30.
- [2] Anantvalee, T., & Wu, J. (2007). A survey on intrusion detection in mobile ad hoc networks. In *Wireless Network Security* (pp. 159-180). Springer US.
- [3] Al-Shurman, M., Yoo, S. M., & Park, S. (2004, April). Black hole attack in mobile ad hoc networks. In *Proceedings of the 42nd annual Southeast regional conference* (pp. 96-97). ACM
- [4] Arora, M., Challa, R. K., & Bansal, D. (2010, April). Performance evaluation of routing protocols based on wormhole attack in wireless mesh networks. In *Second International Conference on Computer and Network Technology* (pp. 102-104). IEEE.
- [5] Bala, A., Bansal, M., & Singh, J. (2009, December). Performance analysis of MANET under blackhole attack. In *Networks and Communications, 2009. NETCOM'09. First International Conference on* (pp. 141-145). IEEE.
- [6] Bakar, K. A. A., & Irvine, J. (2010, September). A Scheme for Detecting Selfish Nodes in MANETs using OMNET++. In *Wireless and Mobile Communications (ICWMC), 2010 6th International Conference on* (pp. 410-414). IEEE.
- [7] Bandeira, N., & Poulsen, L. (2004). U.S. Patent No. 6,728,514. Washington, DC: U.S. Patent and Trademark Office.

-
- [8] Bansal, N., Gupta, S., Dutt, N., Nicolau, A., & Gupta, R. (2004, February). Network topology exploration of mesh-based coarse-grain reconfigurable architectures. In Proceedings of the conference on Design, automation and test in Europe-Volume 1 (p. 10474). IEEE Computer Society.
- [9] Barry, B. I. (2009, March). Intrusion detection with OMNeT++. In Proceedings of the 2nd International Conference on Simulation Tools and Techniques (p. 5). ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).
- [10] Bastian, M., Heymann, S., & Jacomy, M. (2009). Gephi: an open source software for exploring and manipulating networks. ICWSM, 8, 361-362.
- [11] Baumgart, I., Gamer, T., Hübsch, C., & Mayer, C. P. (2011, March). Realistic underlays for overlay simulation. In Proceedings of the 4th International ICST Conference on Simulation Tools and Techniques (pp. 402-405). ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).
- [12] Baumgart, I., Heep, B., & Krause, S. (2007, May). OverSim: A flexible overlay network simulation framework. In IEEE Global Internet Symposium, 2007 (pp. 79-84). IEEE.
- [13] Ben Salem, N., & Hubaux, J. P. (2006). Securing wireless mesh networks. IEEE Wireless Communications, 13(LCA-ARTICLE-2006-009).
- [14] Bicket, J., Aguayo, D., Biswas, S., & Morris, R. (2005, August). Architecture and evaluation of an unplanned 802.11 b mesh network. In Proceedings of the 11th annual international conference on Mobile computing and networking (pp. 31-42). ACM.

-
- [15] Boden, B., Haag, R., & Seidl, T. (2013, October). Detecting and exploring clusters in attributed graphs: a plugin for the gephi platform. In Proceedings of the 22nd ACM international conference on Conference on information & knowledge management (pp. 2505-2508). ACM.
- [16] Boden, N. J., Cohen, D., Felderman, R. E., Kulawik, A. E., Seitz, C. L., Seizovic, J. N., & Su, W. K. (1995). Myrinet: A gigabit-per-second local area network. *IEEE micro*, (1), 29-36.
- [17] Bokor, L., Nováczki, S., Zeke, L. T., & Jeney, G. (2009, October). Design and evaluation of host identity protocol (HIP) simulation framework for INET/OMNeT++. In Proceedings of the 12th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems (pp. 124-133). ACM.
- [18] Bruns, A. (2012). How long is a tweet? Mapping dynamic conversation networks on Twitter using Gawk and Gephi. *Information, Communication & Society*, 15(9), 1323-1351.
- [19] Cetinkaya, E. K., Broyles, D., Dandekar, A., Srinivasan, S., & Sterbenz, J. P. (2010, October). A comprehensive framework to simulate network attacks and challenges. In *Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), 2010 International Congress on* (pp. 538-544). IEEE.
- [20] Chabukswar, R., Sinópoli, B., Karsai, G., Giani, A., Neema, H., & Davis, A. (2010, April). Simulation of network attacks on SCADA systems. In *First Workshop on Secure Control Systems*.
- [21] Chertov, R., Fahmy, S., & Shroff, N. B. (2008). Fidelity of network simulation and emulation: A case study of tcp-targeted denial of service attacks. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 19(1), 4.

-
- [22] Deng, H., Li, W., & Agrawal, D. P. (2002). Routing security in wireless ad hoc networks. *Communications Magazine*, IEEE, 40(10), 70-75.
- [23] Dini, G., & Tiloca, M. (2012, October). ASF: an attack simulation framework for wireless sensor networks. In *Wireless and Mobile Computing, Networking and Communications (WiMob)*, 2012 IEEE 8th International Conference on (pp. 203-210). IEEE.
- [24] Dreibholz, T., Rathgeb, E. P., & Zhou, X. (2009, March). SimProcTC: the design and realization of a powerful tool-chain for OMNeT++ simulations. In *Proceedings of the 2nd International Conference on Simulation Tools and Techniques* (p. 75). ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).
- [25] Ehsan, H., & Khan, F. A. (2012, June). Malicious AODV: implementation and analysis of routing attacks in MANETs. In *Trust, Security and Privacy in Computing and Communications (TrustCom)*, 2012 IEEE 11th International Conference on (pp. 1181-1187). IEEE.
- [26] Fujimoto, R. M., Perumalla, K., Park, A., Wu, H., Ammar, M. H., & Riley, G. F. (2003, October). Large-scale network simulation: how big? how fast?. In *Modeling, Analysis and Simulation of Computer Telecommunications Systems*, 2003. MASCOTS 2003. 11th IEEE/ACM International Symposium on (pp. 116-123). IEEE.
- [27] García-Teodoro, P., Sánchez-Casado, L., & Maciá-Fernández, G. (2014). taxonomy and Holistic Detection of Security Attacks in MAnets. *Security for Multihop Wireless Networks*, 1.

-
- [28] Gorodetski, V., & Kotenko, I. (2002, October). Attacks against computer network: formal grammar-based framework and simulation tool. In *Recent Advances in Intrusion Detection* (pp. 219-238). Springer Berlin Heidelberg.
- [29] Hu, Y. C., Perrig, A., & Johnson, D. B. (2003, September). Rushing attacks and defense in wireless ad hoc network routing protocols. In *Proceedings of the 2nd ACM workshop on Wireless security* (pp. 30-40). ACM.
- [30] Jhaveri, R. H., Patel, A. D., Parmar, J. D., & Shah, B. I. (2010). MANET routing protocols and wormhole attack against AODV. *International Journal of Computer Science and Network Security*, 10(4), 12-18.
- [31] Kannhavong, B., Nakayama, H., Nemoto, Y., Kato, N., & Jamalipour, A. (2007). A survey of routing attacks in mobile ad hoc networks. *Wireless communications, IEEE*, 14(5), 85-91.
- [32] Kaur, R., Sangal, A. L., & Kumar, K. (2014, February). Modeling and simulation of DDoS attack using OMNeT++. In *Signal Processing and Integrated Networks (SPIN), 2014 International Conference on* (pp. 220-225). IEEE.
- [33] Kershenbaum, A., Kermani, P., & Grover, G. A. (1991). MENTOR: an algorithm for mesh network topological optimization and routing. *Communications, IEEE Transactions on*, 39(4), 503-513.
- [34] Kim, H., Kang, I., & Bahk, S. (2004). Real-time visualization of network attacks on high-speed links. *Network, IEEE*, 18(5), 30-39.
- [35] Kotenko, I., & Ulanov, A. (2014). Agent-based simulation of DDOS attacks and defense mechanisms. *International Journal of Computing*, 4(2), 113-123.

-
- [36] Kuhl, M. E., Kistner, J., Costantini, K., & Sudit, M. (2007, December). Cyber attack modeling and simulation for network security analysis. In Proceedings of the 39th Conference on Winter Simulation: 40 years! The best is yet to come (pp. 1180-1188). IEEE Press.
- [37] Kulikowski, K. J., Su, M., Smirnov, A., Taubin, A., Karpovsky, M. G., & MacDonald, D. (2005, March). Delay insensitive encoding and power analysis: a balancing act [cryptographic hardware protection]. In Asynchronous Circuits and Systems, 2005. ASYNC 2005. Proceedings. 11th IEEE International Symposium on (pp. 116-125). IEEE.
- [38] Lau, F., Rubin, S. H., Smith, M. H., & Trajkovic, L. (2000). Distributed denial of service attacks. In Systems, Man, and Cybernetics, 2000 IEEE International Conference on (Vol. 3, pp. 2275-2280). IEEE.
- [39] Lessmann, J., Janacik, P., Lachev, L., & Orfanus, D. (2008, April). Comparative study of wireless network simulators. In Networking, 2008. ICN 2008. Seventh International Conference on (pp. 517-523). IEEE.
- [40] Levchenko, K., Paturi, R., & Varghese, G. (2004, October). On the difficulty of scalably detecting network attacks. In Proceedings of the 11th ACM conference on Computer and communications security (pp. 12-20). ACM.
- [41] Liljenstam, M., Liu, J., Nicol, D., Yuan, Y., Yan, G., & Grier, C. (2005, June). Rinse: The real-time immersive network simulation environment for network security exercises. In Principles of Advanced and Distributed Simulation, 2005. PADS 2005. Workshop on (pp. 119-128). IEEE.
- [42] Magoni, D., & Pansiot, J. J. (2001). Analysis of the autonomous system network topology. ACM SIGCOMM Computer Communication Review, 31(3), 26-37.

-
- [43] Malekzadeh, M., Ghani, A. A. A., Subramaniam, S., & Desa, J. (2011). Validating Reliability of OMNeT++ in Wireless Networks DoS Attacks: Simulation vs. Testbed. *International Journal of Network Security*, 3(1), 13-21.
- [44] Mayer, C. P., & Gamer, T. (2008). Integrating real world applications into OMNeT++. Institute of Telematics, University of Karlsruhe, Karlsruhe, Germany, Tech. Rep. TM-2008-2.
- [45] McNickle, D., Pawlikowski, K., & Ewing, G. (2010). AKAROA2: A Controller Of Discrete-Event Simulation Which Exploits The Distributed Computing Resources Of Networks. In *ECMS* (pp. 104-109).
- [46] McSweeney, P. J. (2009). Gephi Network Statistics. Presentado en Google Summer of Code. Recuperado a partir de <http://gephi.org/google-soc/gephi-netalgo.pdf>.
- [47] Mistry, N. H., Jinwala, D. C., & Zaveri, M. A. (2009). MOSAODV: solution to secure AODV against blackhole attack. *IJCNS) International Journal of Computer and Network Security*, 1(3), 42-45.
- [48] Network Engineering Security Group (NESG): NETA: NETwork Attacks Frame-work for OMNeT++. <http://nesg.ugr.es/index.php/en/neta>
- [49] Riley, G. F. (2003, August). The georgia tech network simulator. In *Proceedings of the ACM SIGCOMM workshop on Models, methods and tools for reproducible network research* (pp. 5-12). ACM.
- [50] Privalov, Alexander Yu, and Alexander Tsarev. "Analysis and simulation of WAN traffic by self-similar traffic model with OMNET++." *Wireless*

Communications and Mobile Computing Conference (IWCMC), 2014 International. IEEE, 2014.

[51] Sánchez-Casado, L., Rodríguez-Gómez, R. A., Magán-Carrión, R., & Maciá-Fernández, G. (2013). NETA: Evaluating the Effects of NETwork Attacks. MANETs as a Case Study. In *Advances in Security of Information and Communication Networks* (pp. 1-10). Springer Berlin Heidelberg.

[52] Varga, A., & Hornig, R. (2008, March). An overview of the OMNeT++ simulation environment. In *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops* (p. 60). ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[53] Yasmeen, F., Ewing, G., Pawlikowski, K. and Yamada, S. 2009 “Distributing Akaroa2 on PlanetLab”. *Proceedings of IEICE General Conference, Matsuyama City, Japan, March 17-20, 2009*

[54] Sterbenz, J. P., Cetinkaya, E. K., Hameed, M. A., Jabbar, A., Qian, S., & Rohrer, J. P. (2013). Evaluation of network resilience, survivability, and disruption tolerance: analysis, topology generation, simulation, and experimentation. *Telecommunication systems*, 52(2), 705-736.

[55] Nilsson, D. K., Larson, U. E., Picasso, F., & Jonsson, E. (2009). A first simulation of attacks in the automotive network communications protocol flexray. In *Proceedings of the International Workshop on Computational Intelligence in Security for Information Systems CISIS'08* (pp. 84-91). Springer Berlin Heidelberg.

[56] Razak, S., Zhou, M., & Lang, S. D. (2002, September). Network intrusion simulation using OPNET. In *OPNETWORK2002 conference*.

-
- [57] Gamer, T., & Scharf, M. (2008, March). Realistic simulation environments for IP-based networks. In Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops (p. 83). ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).
- [58] Gamer, T., & Mayer, C. P. (2009, March). Large-scale evaluation of distributed attack detection. In Proceedings of the 2nd International Conference on Simulation Tools and Techniques (p. 68). ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).
- [59] Guo, J., & Lei, Z. Y. (2011, May). A kind of wormhole attack defense strategy of WSN based on neighbor nodes verification. In Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference on (pp. 564-568). IEEE.
- [60] Jónsson, K. V. (2009, March). HttpTools: a toolkit for simulation of web hosts in OMNeT++. In Proceedings of the 2nd International Conference on Simulation Tools and Techniques (p. 70). ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).
- [61] Qwasmi, N., Ahmed, F., & Liscano, R. (2011, September). Simulation of DDoS attacks on P2P networks. In High Performance Computing and Communications (HPCC), 2011 IEEE 13th International Conference on (pp. 610-614). IEEE.
- [62] Romero, E., Mouradian, A., Blesa, J., Moya, J., & Araujo, A. (2012). Simulation framework for security threats in cognitive radio networks. Communications, IET, 6(8), 984-990.

-
- [63] Díaz, A., Penil, P., Sanchez, P., Sancho, J., & Rico, J. (2012, September). Modeling and simulation of secure wireless sensor network. In *Specification and Design Languages (FDL)*, 2012 Forum on (pp. 185-192). IEEE.
- [64] Sharma, S., & Gupta, R. (2009). Simulation study of blackhole attack in the mobile ad hoc networks. *Journal of Engineering Science and Technology*, 4(2), 243-250.
- [65] Raj, P. N., & Swadas, P. B. (2009). Dpraodv: A dyanamic learning system against blackhole attack in aodv based manet. *arXiv preprint arXiv:0909.2371*.
- [66] Tamilselvan, L., & Sankaranarayanan, V. (2007, August). Prevention of blackhole attack in MANET. In *Wireless Broadband and Ultra Wideband Communications, 2007. AusWireless 2007. The 2nd International Conference on* (pp. 21-21). IEEE.
- [67] da Silva, A. P. R., Martins, M. H., Rocha, B. P., Loureiro, A. A., Ruiz, L. B., & Wong, H. C. (2005, October). Decentralized intrusion detection in wireless sensor networks. In *Proceedings of the 1st ACM international workshop on Quality of service & security in wireless and mobile networks* (pp. 16-23). ACM.
- [68] Weerasinghe, K. G. H. D., & Fu, H. (2008). Preventing cooperative black hole attacks in mobile ad hoc networks: Simulation implementation and evaluation.
- [69] Hermann, C. F., Hermann, M. G., & Cantor, R. A. (1974). Counterattack or Delay Characteristics Influencing Decision Makers' Responses To the Simulation of an Unidentified Attack. *Journal of Conflict Resolution*, 18(1), 75-106.
- [70] Choi, S., Kim, D. Y., Lee, D. H., & Jung, J. I. (2008, June). WAP: Wormhole attack prevention algorithm in mobile ad hoc networks. In *Sensor Networks*,

Ubiquitous and Trustworthy Computing, 2008. SUTC'08. IEEE International Conference on (pp. 343-348). IEEE.

[71] Eui-Jik, K. I. M., Jeongsik, I. N., Sungkwan, Y. O. U. M., & Chul-Hee, K. A. N. G. (2012). Delay attack-resilient clock synchronization for wireless sensor networks. *IEICE TRANSACTIONS on Information and Systems*, 95(1), 188-191.

[72] Chertov, R., Fahmy, S., & Shroff, N. B. (2006). Emulation versus simulation: A case study of TCP-targeted denial of service attacks. In *Testbeds and Research Infrastructures for the Development of Networks and Communities*, 2006. TRIDENTCOM 2006. 2nd International Conference on (pp. 10-pp). IEEE.

[73] Sharma, S., Gupta, R., Reader, M. T. S., SOIT, R., & UIT, R. (2008). Simulation study of blackhole attack in the mobile ad hoc networks. *Executive Development*, 21, 22.

[74] Ghonge, M., & Nimbhorkar, S. U. (2012). Simulation of AODV under Blackhole Attack in MANET. *International Journal of Advanced Research in Computer Science and Software Engineering*, 2(2).

[75] Sharma, N., & Sharma, A. (2012, January). The black-hole node attack in MANET. In *Advanced Computing & Communication Technologies (ACCT)*, 2012 Second International Conference on (pp. 546-550). IEEE.

[76] Purohit, N., Sinha, R., & Maurya, K. (2011, December). Simulation study of Black hole and Jellyfish attack on MANET using NS3. In *Engineering (NUICONE)*, 2011 Nirma University International Conference on (pp. 1-5). IEEE.

[77] Tamilselvan, L., & Sankaranarayanan, V. (2008). Prevention of co-operative black hole attack in MANET. *Journal of networks*, 3(5), 13-20.

-
- [78] Hu, X., Park, T., & Shin, K. G. (2008, April). Attack-tolerant time-synchronization in wireless sensor networks. In INFOCOM 2008. The 27th Conference on Computer Communications. IEEE. IEEE.
- [79] Raniwala, A., & Chiueh, T. C. (2005, March). Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network. In INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE (Vol. 3, pp. 2223-2234). IEEE.
- [80] Chen, T., Zhang, H., Maggio, G. M., & Chlamtac, I. (2007, June). Topology management in CogMesh: a cluster-based cognitive radio mesh network. In Communications, 2007. ICC'07. IEEE International Conference on (pp. 6516-6521). IEEE.
- [81] Wei, H. Y., Ganguly, S., Izmailov, R., & Haas, Z. J. (2005, May). Interference-aware IEEE 802.16 WiMax mesh networks. In Vehicular Technology Conference, 2005. VTC 2005-Spring. 2005 IEEE 61st (Vol. 5, pp. 3102-3106).
- [82] Tang, J., Xue, G., & Zhang, W. (2005, May). Interference-aware topology control and QoS routing in multi-channel wireless mesh networks. In Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing (pp. 68-77). ACM.
- [83] Yao, S., Mukherjee, B., Yoo, S. B., & Dixit, S. (2000, September). All-optical packet-switched networks: a study of contention-resolution schemes in an irregular mesh network with variable-sized packets. In Opticom 2000 (pp. 235-246). International Society for Optics and Photonics.

-
- [84] Grover, W. D., & Stamatelakis, D. (1998, June). Cycle-oriented distributed preconfiguration: ring-like speed with mesh-like capacity for self-planning network restoration. In Communications, 1998. ICC 98. Conference Record. 1998 IEEE International Conference on (Vol. 1, pp. 537-543). IEEE.
- [85] Ramamurthy, S., & Mukherjee, B. (1999, March). Survivable WDM mesh networks. Part I-protection. In INFOCOM'99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE (Vol. 2, pp. 744-751). IEEE.
- [86] Jacomy, M., Venturini, T., Heymann, S., & Bastian, M. (2014). ForceAtlas2, a continuous graph layout algorithm for handy network visualization designed for the Gephi software. PloS one, 9(6), e98679.
- [87] Heymann, S., & Le Grand, B. (2013, July). Visual analysis of complex networks for business intelligence with gephi. In 2013 17th International Conference on Information Visualisation (pp. 307-312). IEEE.
- [88] Heymann, S. (2014). Gephi. In Encyclopedia of Social Network Analysis and Mining (pp. 612-625). Springer New York.
- [89] Hubaux, J. P., Buttyán, L., & Capkun, S. (2001, October). The quest for security in mobile ad hoc networks. In Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing (pp. 146-155). ACM.
- [90] Piro, C., Shields, C., & Levine, B. N. (2006, August). Detecting the sybil attack in mobile ad hoc networks. In Securecomm and Workshops, 2006 (pp. 1-11). IEEE.

-
- [91] Sun, B., Guan, Y., Chen, J., & Pooch, U. W. (2003, April). Detecting black-hole attack in mobile ad hoc networks. In Personal Mobile Communications Conference, 2003. 5th European (Conf. Publ. No. 492) (pp. 490-495). IET.
- [92] Vlieger, E., & Leydesdorff, L. (2011). Content analysis and the measurement of meaning: The visualization of frames in collections of messages. *Public Journal of Semiotics*, 3(1), 28-50.
- [93] Wu, B., Chen, J., Wu, J., & Cardei, M. (2007). A survey of attacks and countermeasures in mobile ad hoc networks. In *Wireless Network Security* (pp. 103-135). Springer US.
- [94] Yang, H., Luo, H., Ye, F., Lu, S., & Zhang, L. (2004). Security in mobile ad hoc networks: challenges and solutions. *Wireless Communications, IEEE*, 11(1), 38-47.
- [95] Zhang, Z., Greiner, A., & Taktak, S. (2008, June). A reconfigurable routing algorithm for a fault-tolerant 2D-Mesh Network-on-Chip. In *Proceedings of the 45th annual Design Automation Conference* (pp. 441-446). ACM.
- [96] Zhou, L., & Haas, Z. J. (1999). Securing ad hoc networks. *Network, IEEE*, 13(6), 24-30.
- [97] Zhu, K., & Mukherjee, B. (2002). Traffic grooming in an optical WDM mesh network. *Selected Areas in Communications, IEEE Journal on*, 20(1), 122-133.
- [98] Medeiros, J. P. S., Brito Jr, A. M., Pires, P. S. M., & Santos, S. R. D. (2009). Advances in network topology security visualisation. *International Journal of System of Systems Engineering*, 1(4), 387-400.

-
- [99] Goyette, R., & Karmouch, A. (2011, July). A Virtual Network topology security assessment. In *Wireless Communications and Mobile Computing Conference (IWCMC), 2011 7th International* (pp. 974-979). IEEE.
- [100] Rossier, C. A., & Germond, A. (1983). *NETWORK TOPOLOGY OPTIMIZATION FOR POWER SYSTEM SECURITY ENHANCEMENT* (No. LRE-REPORT-1983-002).
- [101] Ismail, M., & Sanavullah, M. Y. (2008, December). Security topology in wireless sensor networks with routing optimisation. In *Wireless Communication and Sensor Networks, 2008. WCSN 2008. Fourth International Conference on* (pp. 7-15). IEEE.
- [102] Kim, J., Radhakrishnan, S., & Dhall, S. K. (2004, October). Measurement and analysis of worm propagation on Internet network topology. In *Computer Communications and Networks, 2004. ICCCN 2004. Proceedings. 13th International Conference on* (pp. 495-500). IEEE.
- [103] Markham, T., & Payne, C. (2001, June). Security at the network edge: A distributed firewall architecture. In *discex* (p. 0279). IEEE.
- [104] Ganesh, A., Massoulié, L., & Towsley, D. (2005, March). The effect of network topology on the spread of epidemics. In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE* (Vol. 2, pp. 1455-1466). IEEE.

Appendix

This appendix shows how to import a topology into the Gephi and generate metric analysis results. This graph below is the main UGI of Gephi.

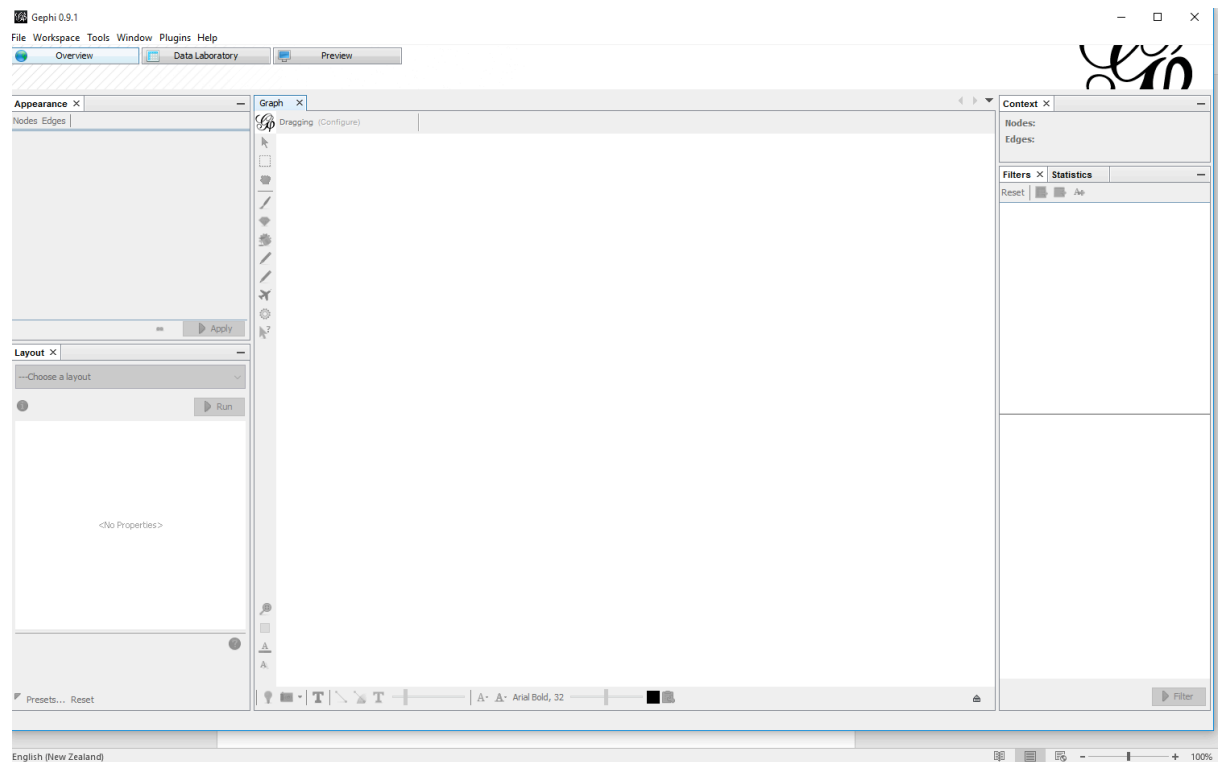


Figure 1: The GUI of Gephi 0.9.1

In the GUI of Gephi: there are main five parts: Appearance, Layout, Graph, Context, Filters and statistics. And there are three operate mode at the top: Overview, Data Laboratory and Preview.

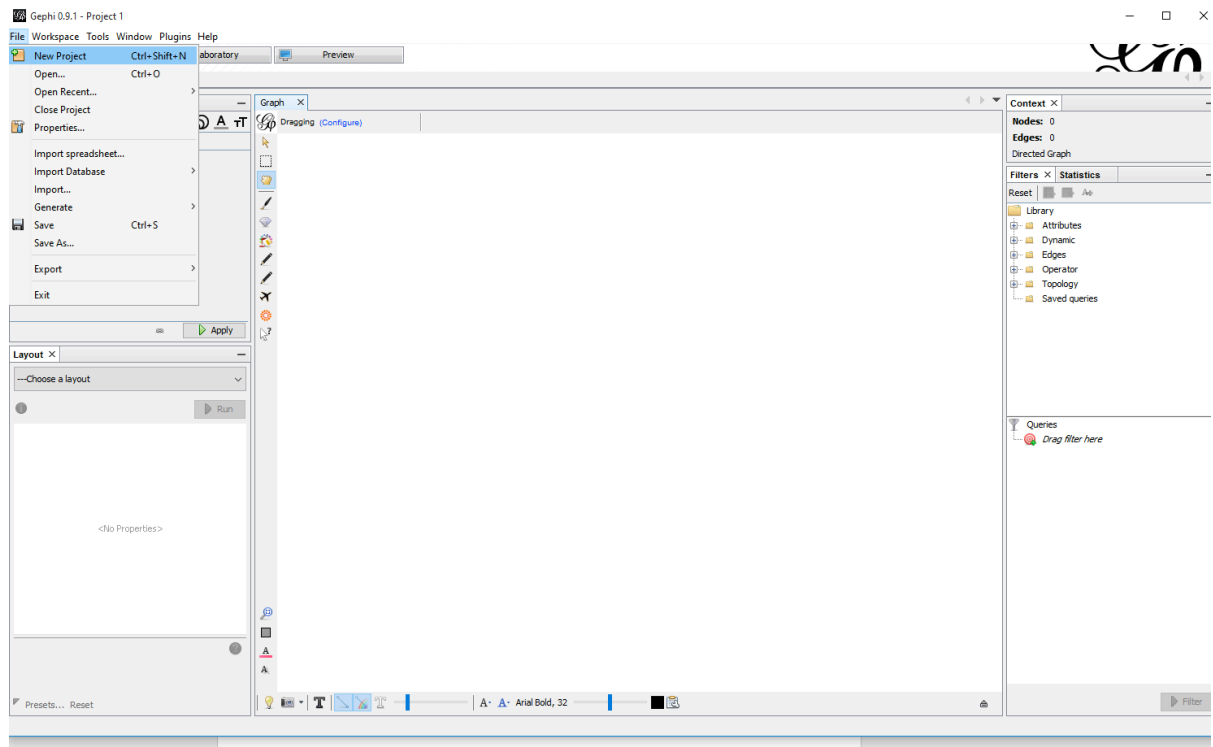
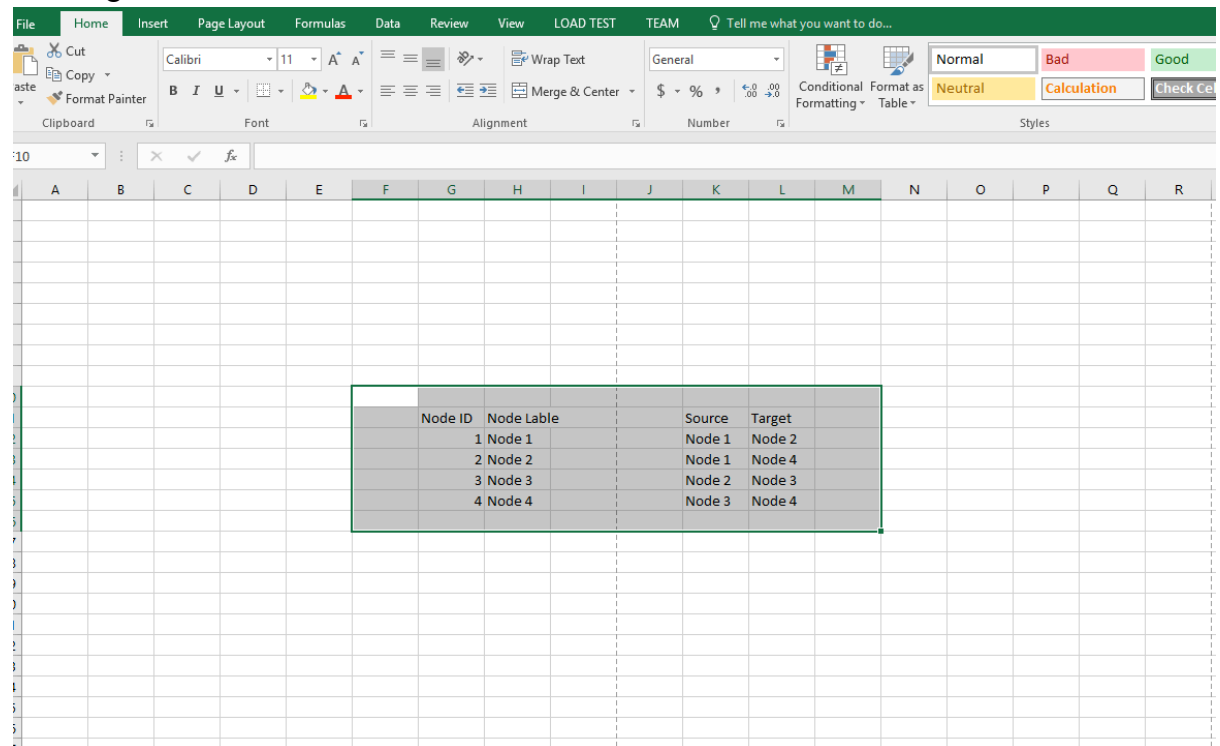


Figure 2: Create a new project in Gephi 0.9.1

The next step is creating a new project by clicking the file icon at the left-top corner. There are two ways to import a topology. One is to use a CSV file to import from outside, and the second way is to enter the node and edges information manually. As an example of this part, we use a simple topology with four nodes and four edges.

1: Using a CSV. File



The screenshot shows the Microsoft Excel interface with the 'Home' tab selected. The ribbon includes options for Clipboard, Font, Alignment, Number, Conditional Formatting, and Styles. A table is visible in the worksheet, spanning columns F to M and rows 1 to 5. The table has the following data:

Node ID	Node Label	Source	Target
1	Node 1	Node 1	Node 2
2	Node 2	Node 1	Node 4
3	Node 3	Node 2	Node 3
4	Node 4	Node 3	Node 4

Figure 3: Edit the CSV File in Excel

There are two files needs to input from the Excel as CSV File, the nodes file and the edges file. As the figure 3 shows, the node s file contain the Node ID and Node Label while in the target file contains Source and Target which shows the forwarding path from node to node.

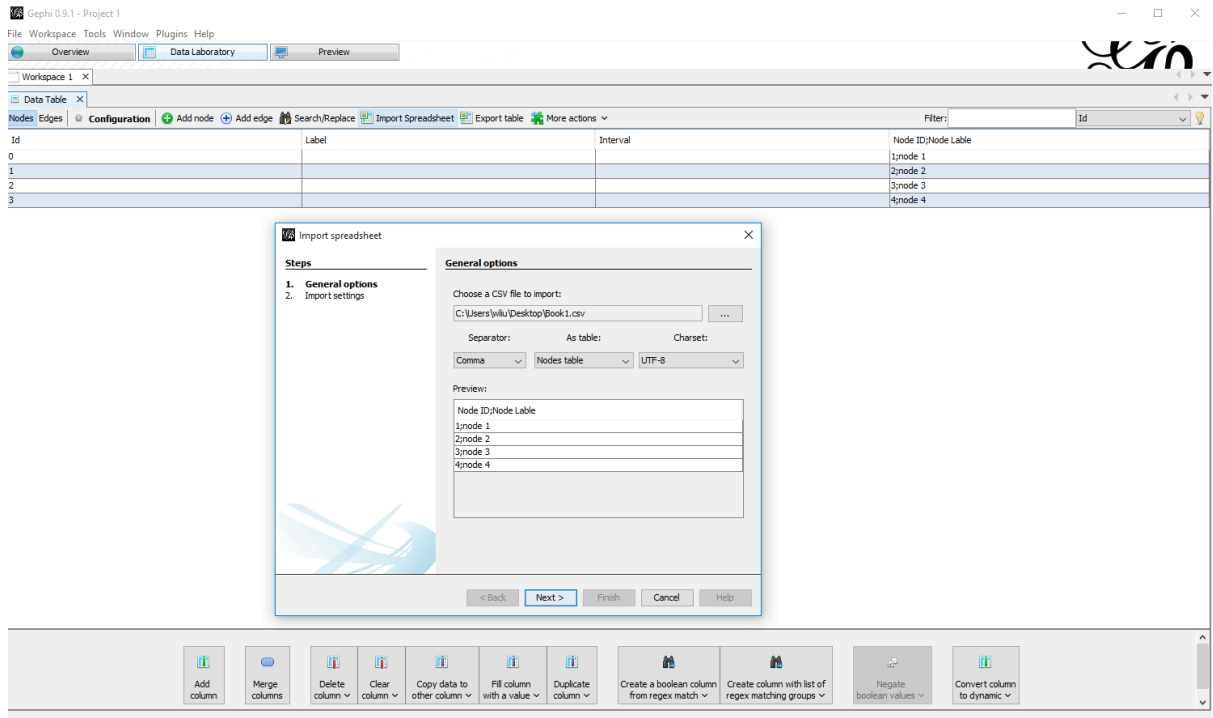


Figure 4: Import the nodes and edges

In the data laboratory, the nodes and edges button is under data tables, and there is a function called import spreadsheet beside that. By click it, you can import the nodes CSV File and edges CSV File.

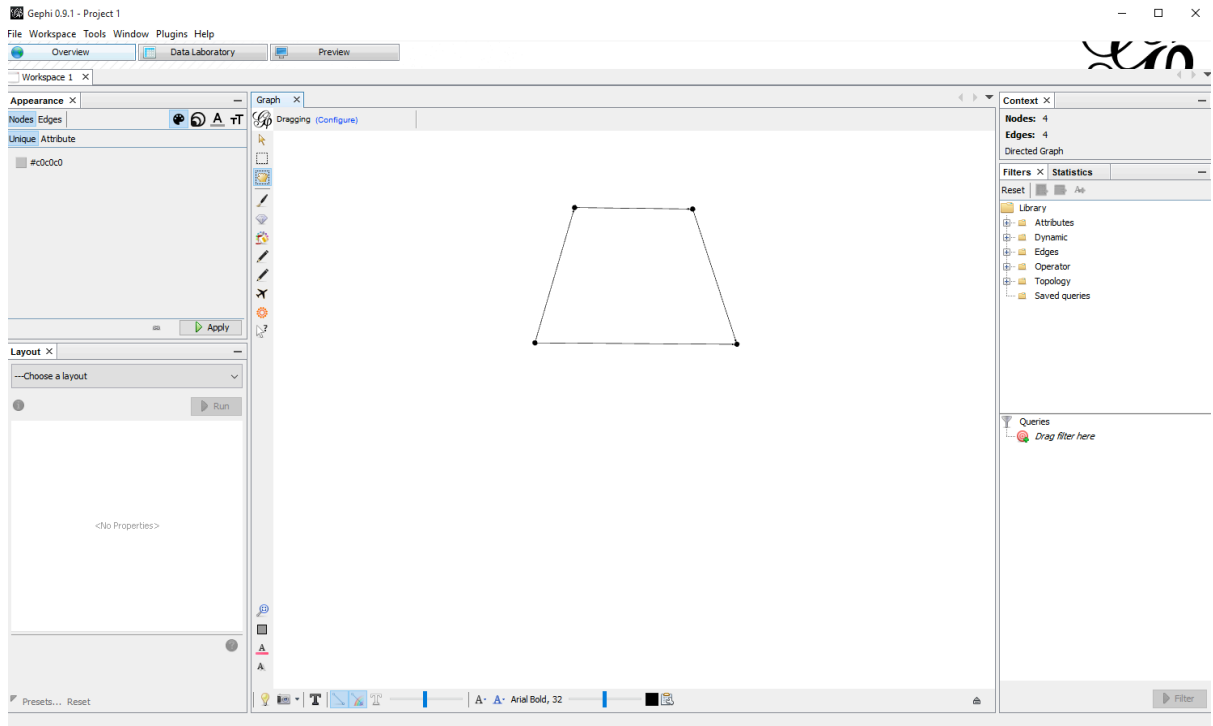


Figure 5: The graph structure of the example topology

When successful import the node file and edge file, the topology graph dragging in the graph panel. As the figure 5 shows, there are four nodes and four links in this topology.

2: Import the topology manually

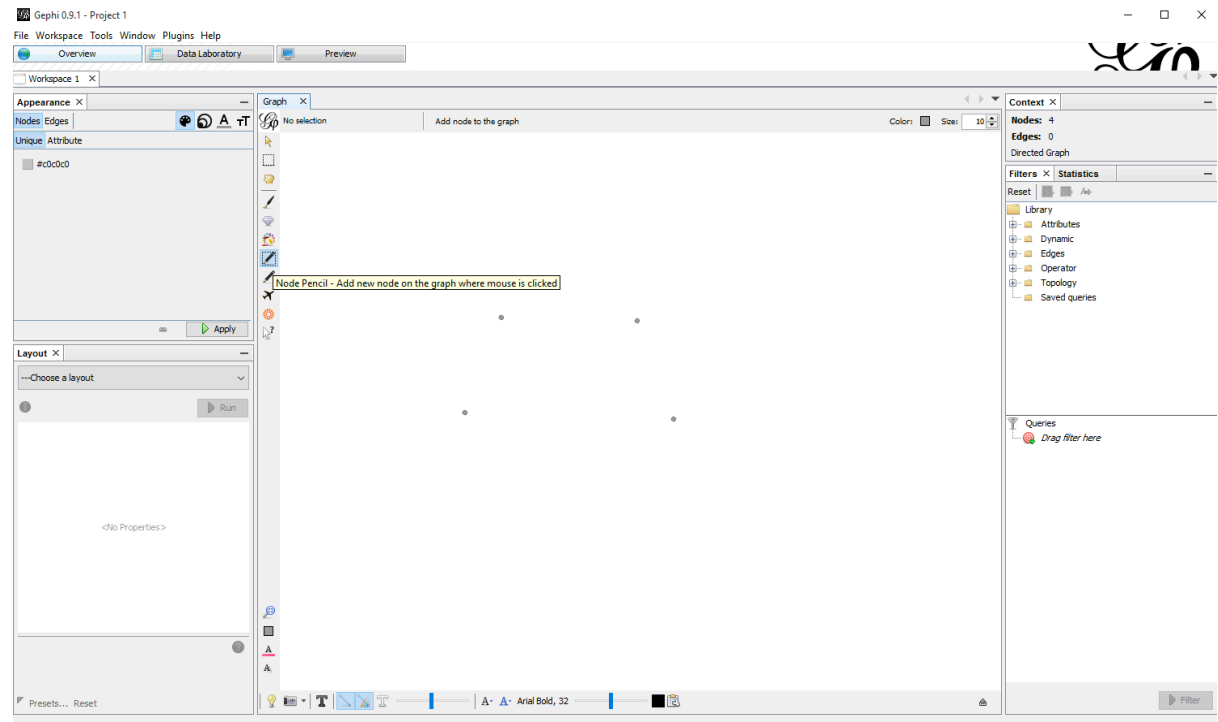


Figure 6: Draw the topology by adding nodes and link edges with the pencil tool

As the figure 6 shows, nodes can be added by using the node pencil as well as the edges can be added by edge pencil. When four nodes and four edges all input by the tools, the graph shows as follow:

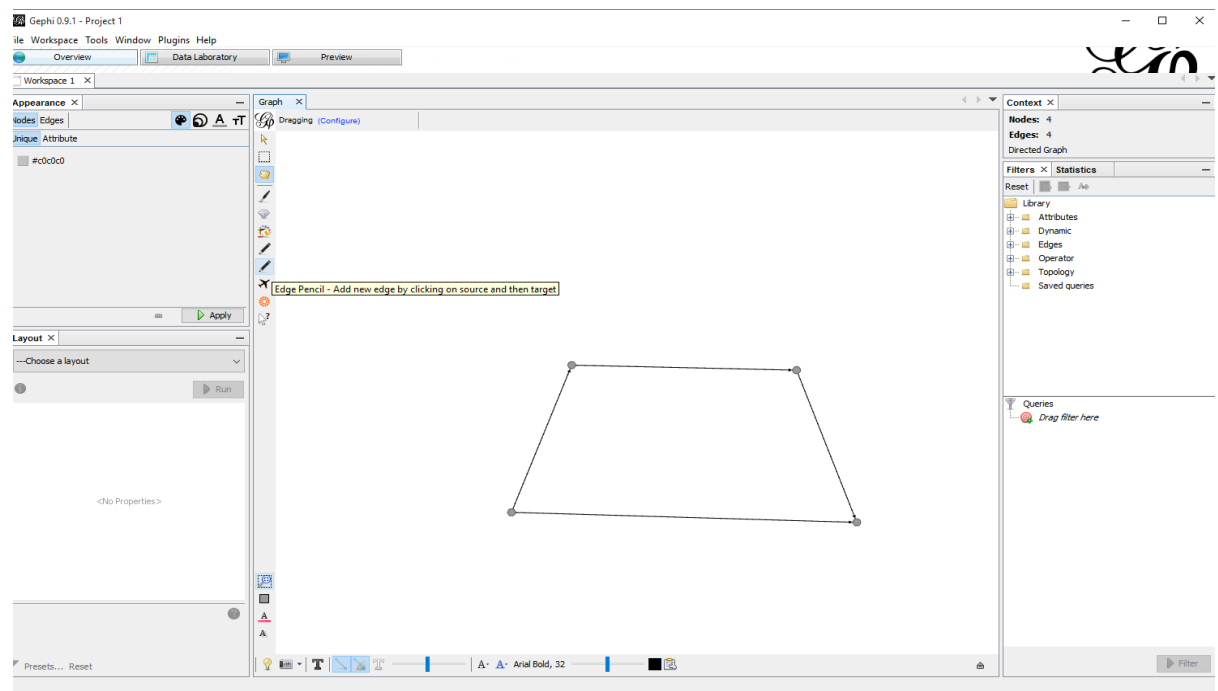


Figure 7: The example topology structure import manually

When the topology is imported into the Gephi, the next step is to analysis the topology by using the statistics function.

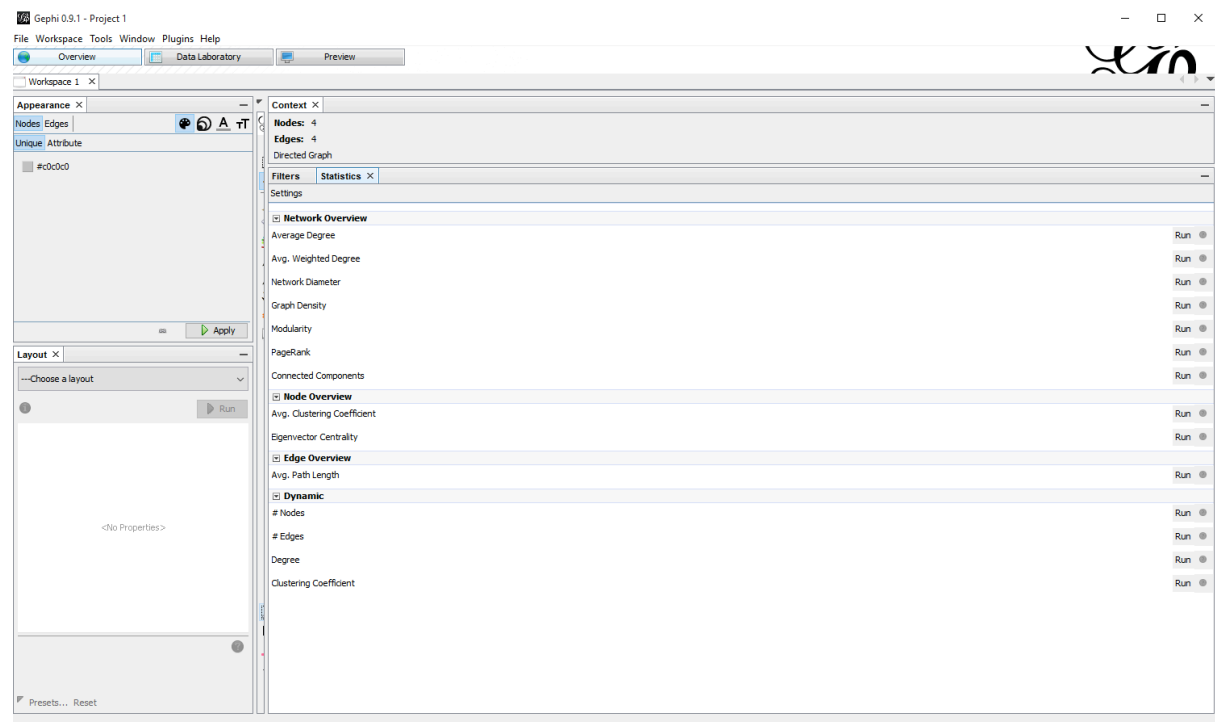


Figure 8: The statistics function of Gephi 0.9.1

As the figure 8 shows, there are four analysis part in the statistics function which are Network Overview, Node Overview, Edge Overview and Dynamic. In the network overview, there are seven metrics which are average degree, average weight degree, network diameter, graph density, modularity, pagerank and connected components. The average clustering coefficient and eigenvector centrality are located in the node overview while the path length is in the edge overview. Lastly, the nodes, edges, degree and clustering coefficient are in the dynamic part.

As the network diameter metric as an example, when running the network diameter, a graph distance report is generating and shows analysis result like betweenness centrality distribution and some other evaluation graph.

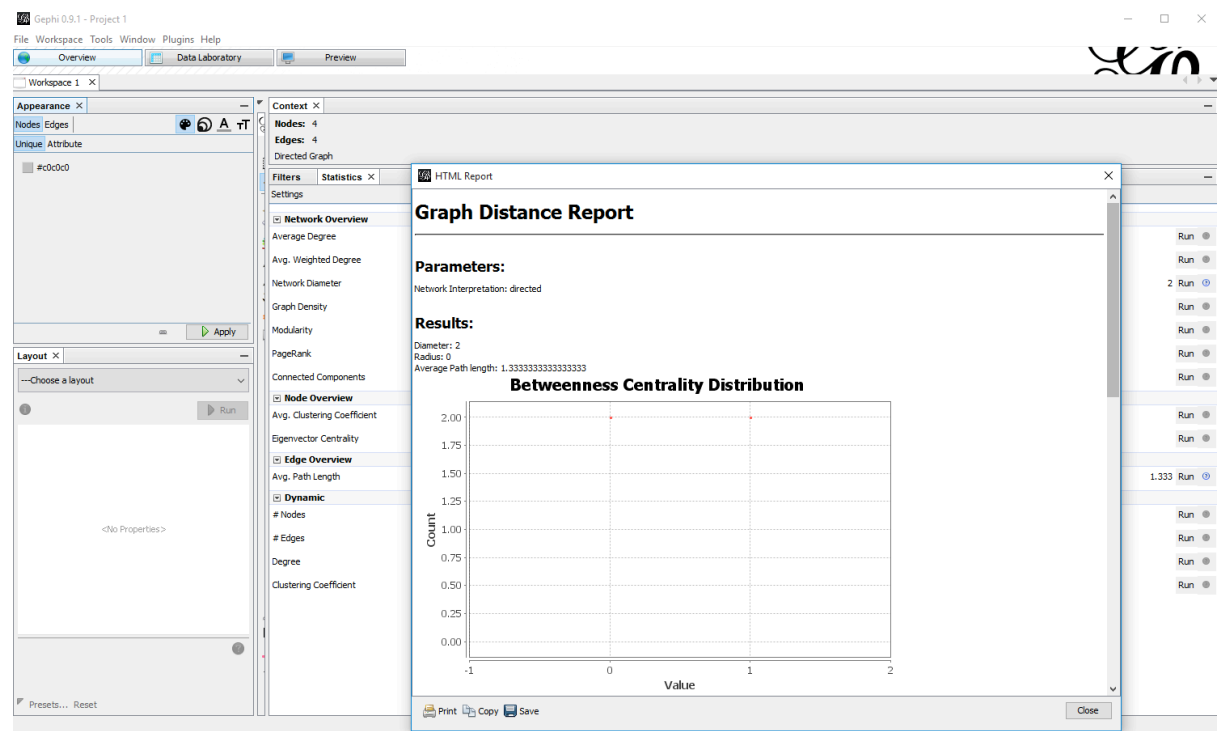


Figure 9: The analysis report of network diameter as an example

This appendix can be used as a simple tutorial to show how to import the topology into the Gephi and generate the analysis results.
