KWAME NKRUMAH UNIVERSITY OF SCIENCE AND TECHNOLOGY, KUMASI

COLLEGE OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE

IMPROVING NETWORK SECURITY BY SANITIZING NETWORK TOPOLOGICAL INFORMATION TO ENCAPSULATE PARTICULAR NETWORK TOPOLOGY

A THESIS SUBMITTED TO THE DEPARTMENT OF COMPUTER SCIENCE IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD

OF

MASTER OF SCIENCE DEGREE IN INFORMATION TECHNOLOGY

By

JEFF GBATE

JUNE, **2019**

DECLARATION

I, Gbate Jeff, hereby declare that this submission is my own work towards the MSc and that, to the best of my knowledge, it contains no material previously published by another person nor material which has been accepted for the award of any other degree of the University, except where due acknowledgment has been made in the text.

GBATE, JEFF (PG8963413)	……	…….
(Student Name & ID)	(Signature)	(Date)

Certified by:

DR. M. Asante	……	…….
(Supervisor(s) Name)	(Signature)	(Date)

Certified by:

DR. M. Asante	……	…….
(Head of Department)	(Signature)	(Date)

Abstract

Securing data is crucial in a world where attackers will attempt to gain access to personal and business information that, for privacy reasons, we want to protect. Information on a network topology can be used by an adversary to undermine a network. To preserve this information, access to the data can be restricted. What if, however, we want to share the information with another party to allow analysis on the data? How can we ensure that the privacy of our data is protected while still providing accurate analysis? Summarizing the information of network topology may not allow for any analysis to be performed on the data. Sanitization, on the other hand, explores methods to mask the network topology information in such a way that the network's characteristics will be hiding while still providing an analyst with data on which she can run statistics. There exists a tenuous balance between the need for privacy of the unsanitized network data and the accuracy of the analysis on the sanitized data. The goal is to effectively hide the raw data while the analysis on the sanitized data produces the same results as if performed on the raw data. By exploring the network characteristics, we can determine possible statistics that may derive from the data. We proposed a standard syntax for defining a network. From the syntax and the topology characteristics, sanitization schemes are detailed explaining the balance between privacy and statistical accuracy. IP addressing ramifications and the concerns when sanitizing are also identified.

TABLE OF CONTENT

Contents Page

LIST OF TABLES

LIST OF FIGURES

CHAPTER ONE

INTRODUCTION

1.0 Introduction

The expanding requirements for computer access and openness to data, PC network and areas have turned out to be more beneficial. Associations rely on a network to give communication access, information access and the exchange of data between units.

Associations and Individuals use their systems for a wide range of capacities that are from overseeing and treatment of data. Thus, the ability to get to this data rapidly and effortlessly is not kidding to the Organizations.

In the meantime, Organizations wish to restrain this entrance to just licensed individuals. They may depend on the mystery of the network design as a first obstruction towards keeping impostors or trespassers out.

For this reason, Organizations wish to conceal data about their network topologies and foundation and additionally data about the frameworks associated with that network.

In this view, this proposal delivers the need to keep up the security of an association's network by sanitizing the network topological data to enable the information to be shared and examined by different entities.

1.1 Background to the Study

Protection is withholding data that you would prefer not to impart to another element or unit. This substance or unit can be inside or outer to your association. To impart data to another substance or unit, this sharing can be in one of these two structures:

- Summary analysis of the raw data or
- Sanitization or Refining of the raw data.

Summary analysis of the raw information is a factual examination of the crude information. This examination can incorporate record checks, average, minimums and maximums, and so forth. For a peer-to-peer network, for instance, one can condense the data of system topology, for example, the number of nodes, the number of connections and the average number of associations with a node.

Sometimes investigation of the raw information happens before the information is imparted to another substance or providing for the examiner to work on or dissected. On account of this the beneficiary of the information does not approach the correct unique raw information and in this way can't reproduce this original information.

In any case, with summarized information, the examination displayed is all that is accessible to the beneficiary. He or She cannot make any insights except for those resultant from the summarized information since he or she do not have access to the raw information. The summary is static and the beneficiary cannot cooperate with it like he or she would have the capacity to on the off chance that he or she approached the raw information.

Sanitization or Refining, then again, furnishes an element with the raw information. The information is controlled such that, the sensitive part of the information that the proprietor wishes to keep private, is covered up or secured. The benefit of this approach is that the beneficiary of the information can examine the information and make his own measurements however the test with this technique is the procedure by which the sensitive information is covered up or secured. On the off chance that this is done inadequately, the beneficiary might have the capacity to recreate the first raw information or may get himself unfit to figure the coveted measurements utilizing the sanitized or refined information. In the case of a system's node names, each name may be replaced with another irregular identifier, along these lines concealing the node's actual name from the beneficiary of the raw data.

1.2 Statement of the problem

Keeping network topology data private may not be sensible for the more significant part of the associations or each association. Time and costs must be put into, to ensure the protection of this information. Likewise, new arrangements and the authorization of these strategies together with changes to existing approaches should be considered. It might be valuable to impart data to some outside gathering about the network, without uncovering data about the topology that the association wishes to keep private. The association could give network topology data that has been separated or adjusted such that the information can utilize without uncovering excessively about certain topological qualities inside the network. The need to keep up the security of network topology while at the same time keeping up the capacity to examine its network data might be indispensable to each association's prosperity.

In summary, the issue is to conceal the connection between the nodes on the network. Each network topological data was purified (sanitized) to keep it from being accessed either specifically from the sanitized information or by reproducing the first network topology utilizing the sanitized data.

1.3 Purpose of the study

As technology (networking) integrates into every aspect of business practices, there is the need for businesses to protect against, scamming and skimming. But who would think to protect against the network information that is supposedly already sanitized?

The objective of this work is to think about the sanitization of network data and to see how to adjust secrecy with the requirement for required investigation and furthermore to introduce diverse strategies for sanitization that an information proprietor can use to successfully cover up various parts of their network topology data from an enemy. An enemy must be kept from inducing with

the sanitized information. Our examination will decide the adequacy of every strategy and under what assumptions the crude information can be recovered or recreated from the sanitized information.

This study aim at:

- Effectively cover up various parts of the system (network) topological data from an enemy.
- Prevent the enemy from deriving the unsanitized information from the non-sanitized parts of the information
- Determine the adequacy of every strategy and under what assumptions the unsanitized (crude) information can regain from the sanitized information.

1.4 Research Questions

1. How can sanitization help in privacy on the network topological information?
2. What kind of private data or network topological information can help an adversary
3. To reconstruct the original data from the sanitized one?
4. What type of sanitization techniques can be implemented to hide the network topological information?

1.5 Significance of the Study

Since this problem is not only strange to organizations, but also educational institutions, individuals, government bodies and Businesses, the whole nation will benefit from this research. As a result, it will help to improve and keep up protection of specific parts of their system while at the same time sanitizing system data to enable their information to be shared and investigated by different substances.

Also, it will help maintain the privacy of network topology information which adversary uses some time to reconstruct unique information from the sanitized data while at the same time yet keeping

up the capacity to be able to examine and share their network data which is vital to the accomplishment of their associations.

1.6 Limitations of the Study

It would have been of good to increase the sample size of the study to cover the entire network topology as a whole, but that was not possible due to financial resources and time duration at my disposal as the researcher.

1.7 Definition of Terms

Computer Networking: A computer network or data network is a telecommunications network that computers use to exchange data. In computer networks, networked computing devices pathway for data to each other along data connections as shown in Fig1.1

Fig1.1 Sample of a computer networking.

A network as portrayed can be ordered utilizing three properties: convention (protocol), design, and topology. The assembly of a system alludes to the necessities, outline, or plan to depict how the equipment (hardware) and programming (software) inside the system deployed. Normal

composes are shared (peer-to-peer) and client-server. The focus of this research, however, would be on the topology of a network hence the need to illustrate what a system topology is, and afterward making a formal language structure (syntax) to show the topology.

Network Topology: Network topology is the physical arrangement of the various parts (links, nodes, etc.) of a computer network. Mostly, it is the topological structure of a network and may depict physically or logically. Computer network topology is the organization of the various element (nodes, links, peripherals, etc.) of the network.

Network topologies look not only at the physical but also the logical aspect. The topology of the network determines the way in which different systems and nodes are connected and communicated with each other. Physical Topology is the physical structure of nodes, workstations, and cables in the network as shown in the Fig1.2, while logical topology is the way information flows between different components.

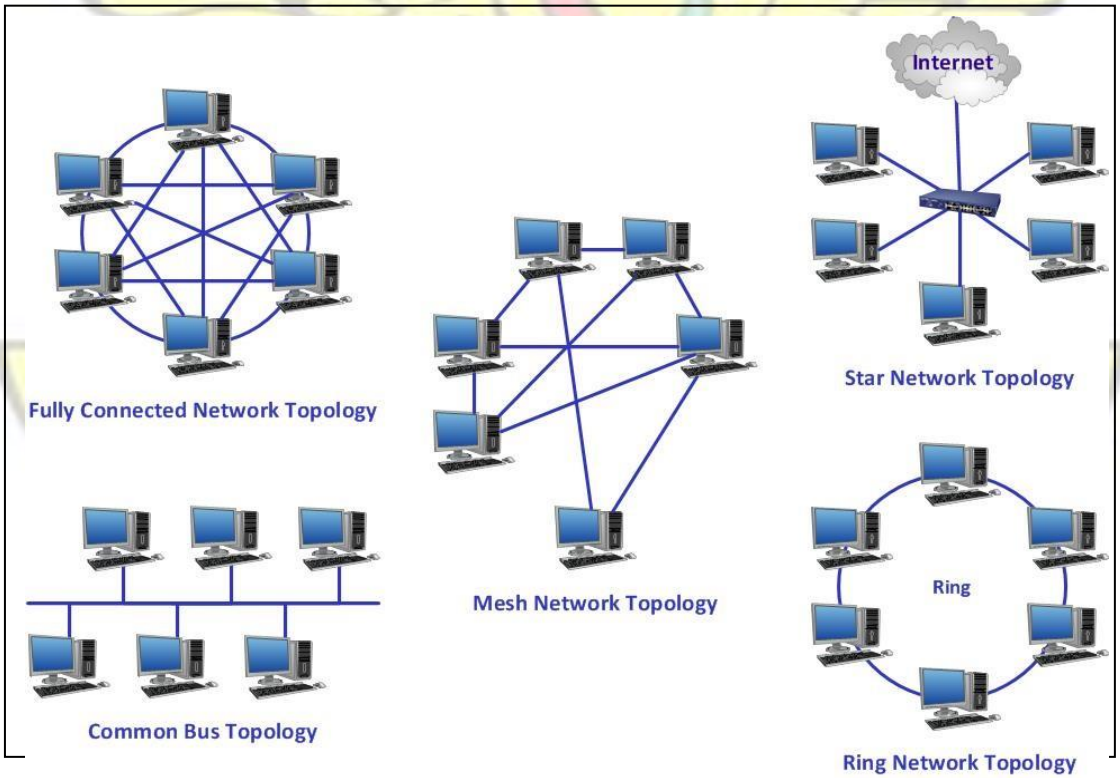


Fig1.2 Examples of **network topology (**physical layout of nodes).

1.8 Thesis Organization

This proposal is sorted out into five chapters. Chapter one presents the topic of the theory and the setting in which it connected.

Chapter two is about the literature review; it focuses on what other publications or researchers have said about the problem on which the researcher wants to solve or improve.

Chapter three describes a penetration test performed and the use of a standard syntax. And also discuss the methods to sanitize network topological information and other countermeasures.

Chapter four describes the results every strategy has on the network's qualities and investigation of the system and furthermore how the sanitization strategies can be applied to a complex network example as well as identifies resiliency issues in networking.

Chapter five summarizes the results and will also give recommendations and suggestions for further research.

CHAPTER TWO

LITERATURE REVIEW

2.0 Introduction

Currently, thanks to the stunningly fast advancement of the computer and network technology, people can quickly send or receive secret information in various forms to or from almost any remotest part of the world through the Internet within seconds. There might be lots of confidential information being transmitted and exchanged on the Internet at this particular point in time. However, important confidential messages may be leaked while they are transferred over some common communication channel. Therefore, achieving safe secret communication is an important field for researching.

Normally, a cryptographic technique such as DES or RSA is used to encrypt the private message before it is sent out through the Net, to make it appears meaningless.

Though modern cryptographic systems offer quite high a security level to the secret information transmitted and exchanged on the Net, the confidential information's appearances as ciphertexts readily draw hackers' attention as though proclaiming that some secret messages of exceptionally high value could obtain if the ciphertexts could only be in one way or another decrypted. Under such circumstances, the hackers are probably not able to decrypt the ciphertexts due to the strong security of the cryptographic system, at least the irritated attackers can destroy the ciphertexts and make the transmissions fail. As a result, it seems that of ensuring the security of secret information traveling on the Internet is the best policy to avoid any attention and suspicion of the hacker. That is to say, the safest method to keep messages transmitted through open channels from leaking out is to encrypt them into a meaningful content

(i.e., plaintext), and this is where steganography, data hiding comes into play.

Given addressing this problem, many few researchers have done many good works to help treated or reduce the situation on the ground by using:

1. Deception
2. Data sanitization
3. Data hiding using: A. Steganography
   B. Cryptography

2.1 DECEPTION

Hiding things from hackers is standard practice in computer security. Normally, systems and hidden files behind firewalls and access-controls, and some data are protected by encryption. By denying information to hackers, these common forms of idiotypically work. Deception is another way to hide things. Deception, currently, is an emerging and promising means for computer security, as seen with honeypots (Spitzner, 2003). Therefore, this work looks at hiding things from hackers using deception as a means. A wide variety of computer security applications used deceptive hiding. One such application involves hiding information about a network 's topology, vulnerabilities, and assets from hacker survey (for example, scanning). The honeypot, for example, intercepts connections to unused network addresses and impersonates computers at those addresses (Spitzner, 2003). Its use makes it difficult for hackers to find real computers and to scan the network without being detected.

Deception was being used to hide computer-security devices, including firewalls, intrusion, detection systems, keystroke loggers, and honeypots. For example, a firewall can send fake

ICMP ‘ host unreachable’ messages in response to disallowed packets, making it look that the firewall, and victim computers behind it, are not on the network.

Furthermore, deception computer security aims at a predictable course of action or inaction that can be exploited or otherwise used to advantage by a hacker to mislead a hacker (Dewar, 1989). In general, activities that cause the hacker to act dangerously or unpredictably should avoid. For example, suppose a system administrator hides network logs to prevent hackers from deleting their routes. The entire hard drive may be erased by a hacker in other to be safe if the expected records are missing, Therefore, is anticipating such unintended consequences and taking actions to mitigate their effect is an essential aspect of deception planning.

Things hide from an agent, human or computer in the context of computer security. The agent whom the item is being protected from will be known as the target. The target is a hacker or a hacker’s automated agent (for example, a worm). For deception operations, in general, the adversary who is being deceived is referred to as the deception target. For deceptive hiding, the goal of hiding is also the deception target.

This paper explains how deceptive hiding works regarding whereby it misrepresents, or knowhow, a single target (hacker). However, improving computer security in some specific way is the deception planner’s ultimate idea but not lying to the target. Deception’s fraud can be both alluring and intriguing, making it easy to lose sight of the deception’s ultimate purpose.

Deception is a make of perception in which a target is purposely led to an incorrect thought, through the actions of another (Whaley, 1982). Deception is distinguishing from unintentional acts of misrepresentation and self-induced acts of deceit (self-deception). Bell and Whaley categorize deceptions as hiding and showing (Bell et al., 1982). Deceptive hiding conceals or obscures a thing’s existence or its attributes in a way that intentionally misleads the target. It is distinguished

from denial, which may also involve hiding, but without the intent to mislead. Denial withholds merely information from the target. Encryption is an example which openly covers a message but not its existence. Steganography, which aims to hide the existence of communication, on the other hand, is deceptive, as it uses in misleading data carrier (for example, writing is hidden in the low-order bits of an image file in such way that the text is not visible to the naked eye).

In deceptive showing makes something that does not exist appear as if it does by portraying one or more of its attributes. For example, after several unsuccessful logins, a computer can continue to prompt for passwords, but ignore them and not permit login. The computer is deceptively showing login prompts. Hiding and showing are both present in any act of deception (Bell and Whaley, 1982). When showing the false, the truth must also be hidden. When something is hidden, something else is shown instead, even if only implicitly. Further, deceptions are often constructed of multiple cons, employing both hiding and showing. For example, a honeypot can deceptively impersonate (that is, show) a network server, while deceptively hiding a keystroke logger. When a deception uses both hiding and showing, the deception may be characterized as hiding or showing, according to the planner's primary intent.

For instance, a server 's banner according to Bell and Whaley is adjusted to expose a false model and version number. The flag is showing the lie, but the principal intent is hiding from hackers and worms the server's true model and version.

Bell and Whaley based on three ways of hiding offer a taxonomy of deceptive techniques namely: masking, repackaging, and dazzling; as well as three ways of showing namely: mimicking, inventing, and decoying (Bell and Whaley, 1982). The military and computer security literature used the taxonomy (USMC 1989, Julian 2002). The military deception literature also lists common types of battlefield deceptions, examples being camouflage, feints (fake attack

introduction), ruses (tricks made to deceive), demonstrations (fake force deployment), and displays (the showing of fake military forces or equipment, for example, inflatable tanks) (U.S. Army 1998, Dewar 1989, Fowler and Nesbit 1995). Cohen (1998) and Rowe and Rothstein (2004) have shown how these can be applied to computer network defense. Rowe and Rothstein also give a taxonomy of deception techniques based on semantic cases in computational linguistics such as agent, instrument, location-from, time-at, and purpose. Also, Rowe has developed a taxonomy for deception in virtual communities (Rowe,2005). The taxonomy applies primarily to computer misuse, and not to computer security.

2.2 DATA SANITIZATION

According to Dale Edgar (2000) defines Data Sanitization as the process of disguising sensitive information in text and development databases by overwriting it with realistic looking but false data of a similar type. It was deduced from Dale that protect valuable business information data in testing environments should be sanitized and also it is being ensured by a legal obligation in most countries, and in protecting valuable information came out with two fundamental type of security. The initial type is concerned with the integrity of the data where the modification of the records is precisely controlled. For example is an account to be credited or debited without specific controls and auditing.

The second type of security by Dale Edgar (2000) is the protection of the information content from inappropriate visibility where he uses Names, addresses, phone numbers and credit card details as good examples of this type of data. Unlike the protection from updates, he starches that, this type of security requires that access to the information content should control in every environment.

The idea behind data sanitization was introduced in Atallah et al. (1999) considering the problem of modifying a given database so that the support of a given set of sensitive rules decreases below the minimum support valve. Atallah et al. (1999), focuses on the theoretical approach and showed that the optimal sanitization is an NP-hard problem.

Secondly, Atallah et al. (1999) investigated confidentially issues of a broad category of associated rules and proposed some algorithms to preserve the privacy of such rules above a given privacy threshold.

In the same direction, Saygin et al. (2001) introduced some algorithms to obscure a given set of sensitive rules by replacing know valves with unknowns, while minimizing the side effects on non-sensitive rules.

Oliveira and Zaiane (2003) in sanitizing data introduced a framework for protecting restrictive patterns composed of sanitizing algorithms that require only two scans over the database. Here in this framework, the first scans by Oliveira and Zaiane (2003) are required to build an index (inverted file) for speeding up the sanitization process while the second scan is used to sanitize the original database.

Legal Obligations

The legal conditions for Data Sanitization differ from country to country, but most countries now have laws of some form. Examples are:

United States: The Gramm-Leach-Bliley Act wants institutions to protect the confidentiality and integrity of individual customer information. The Right to Financial Privacy Act of 1978 formulates legal Fourth Amendment protection for financial reports, and there is a multitude of different state laws. There are also some security and privacy requirements for personal information included in the Health Insurance Portability and Accountability Act of 1996 (HIPAA)

The European Union: Directive 95/46/EC of the European Parliament which gives strict guidelines concerning individual rights to data privacy, and the duties of data handle to secure against misuse.

The United Kingdom: The United Kingdom Data Protection Act of 1998 spreads the European Parliament directive and sets further lawful obligations on the handlers of individual, private or sensitive data.

As with most things legal, the features are open to discussion. Though, if data for which your organization is responsible gets loose and appropriate steps were not taken to prevent that release, then your organization's lawyers will find themselves in court trying to put their best spin on the matter. However large the legal liabilities are, they could seem trivial in comparison to the losses associated with the disastrous loss of business confidence caused by a large-scale privacy breach.

Any business that outsources test and expansion operations needs to be very mindful of the specific laws governing the transmission of information across national borders.

2.2.1 Data Sanitization Techniques

One needs to work with databases when testing and development teams which are structurally suitable working copies of the real environments. However, they do not surely need to be able to see security sensitive information. As long as the data looks real, for test and development purposes, the real record content is usually unnecessary.

There is a type of Data Sanitization techniques available – the pros and cons of some of the most useful are considered below.

Technique: NULL'ing Out

According to Dale Edgar (2000), it is a simply deleting a column of data by replacing it with NULL values. Here he stated an example saying, it is very hard to write and text customer account maintenance forms if the customer's name, address and contact details are all null values.

The NULL'ing Out technique is useful in certain specific circumstances but rarely used as the entire Data Sanitization strategy.

Technique: Masking Data

Masking data means replacing certain fields with a Mask character (such as an X), according to Dale Edgar (2000), this effectively disguises the data content while preserving the same formatting on front end screens and reports, for example, uses a column of credit card numbers which might look like: 4346 6454 0020 5379

4493 9238 7315 5787 4297 8296 7496 8724 and after the

masking operation the information would appear as:

4346 XXXX XXXX 5379

4493 XXXX XXXX 5787

4297 XXXX XXXX 8724

Which still preserving the look and feel effectively remove much of the sensitive content from the record. To preserve security one must take care to ensure that enough of the data is masked. It would not be hard to reconstruct the real credit card number from a masking process beforementioned as 4297 8296 7496 87XX since the figures are created with a specific and well-known checksum algorithm.

Also, A masking operation such as XXXX XXXXXXXX 5379 care must be taken not to mask out potentially required information card issuer details from the credit card number. This may, or may not, be pleasing.

From Dale Edgar, Masking is a powerful and fast Data Sanitization option if the data is in a specific, invariable format. Masking can be slow, extremely difficult to administer and can potentially leave some data items inappropriately masked If numerous special cases must be dealt with.

Technique: Encryption/Decryption

Dale Edgar (2000) also looked at Encryption/Decryption and finalized that, this technique gives the opportunity of dropping the data in place and obvious to those with the proper key while prevailing effectively worthless to anyone without the passkey.

This would appear to be a very good choice – yet, as with all techniques, it has its strengths and weaknesses.

Encryption according to him, also destroys the formatting and look and feel of the data. Encrypted data rarely looks meaningful; in fact, it usually looks like binary data. This sometimes leads to NLS character set issues when manipulating encrypted varchar fields. Certain types of encryption impose constraints on the data format as well. For example, the

Oracle Obfuscation toolkit requires that all data to be encrypted should have a length which is a multiple of 8 characters. In effect, this means that the areas must be expanded with a suitable filling character which must then be removed at decryption point.

Some encryption is more secure than others, which makes the strength of the encryption also an issue. And just a matter of time and effort most encryption systems can be broken. However, not very much will keep the national security companies of largish nations from viewing your files should they choose to do so. This may not be a big worry if the requirement is to protect proprietary business information.

The security is dependent on the strength of the encryption used. It may not be suitable for highsecurity requirements or where the encryption key cannot be secured. Encryption also destroys the look and feel of the sanitized data. The big plus is the selective access it presents by Dale Edgar (2000).

2.3 Data Hiding

As the Internet permeates our daily lives, there is a need to address issues of protection; flexible security for evolving network applications is required. This work attempts to integrate traditional network security with another emerging technology, data hiding. Many forms of information hiding such as encryption are used for data hiding, where both parties encrypt the information and transfer a cipher. These techniques have become much more open and public in the last few years.

The steganography aims to prevent a third party from realizing that any covert communication has taken place better than the encryption Steganography is defined as the art and science of hiding information, transmitting secret messages through innocuous cover carriers in such a manner that the existence of the embedded messages is undetectable. To be able to decode and view the information, only persons who have knowledge of the embedded information and possess a "key," which can use many kinds. It can cover from a passphrase for electronic steganography to a compromise of a method to decode the information.

2.3.1 Steganography:

We are going to discuss steganography in this paper; it is defined as the art and science of hiding information, which is a process that involves hiding a message in an appropriate carrier, for example, a text file. The transport can then be sent to a receiver without anyone else knowing that it contains a hidden message. Steganography is a common term referring to all methods for the embedding of additional content into some form of a carrier the choice of the carrier is nearly endless; it may be an old piece of parchment, as well as a network protocol header. Present day steganographic methods are far more complex than their old predecessors, but the main laws had remained stable. The utilization of digital media data or network protocols as a carrier is what they typically rely on in which secret data is fixed. All methods for the embedding of additional secret content into some form of carrier, with the aim of concealment of the introduced alterations is refer to as Steganography. The decision of the transport is nearly endless; it may be an ancient piece of parchment, as well as a network protocol header. Inspired by biological phenomena, adopted by man in the ancient times, it has been developed over the ages.

2.3.2 Cryptography:

Cryptography is a process of saving and transferring data in a form so that it can no more be interpreted or understood. It is a science of protecting an effective way of protecting sensitive information as it is stored on media or transmitted through network communication paths

A cryptographic algorithm, roughly cipher, is a mathematical capacity used in the encryption and decryption method. A cryptographic algorithm operates in blending with a key a word, figure, or expression to encrypt the plaintext the same plaintext encrypts to another ciphertext with another key. The security of encrypted data is uniquely dependent on two things: the strength of the cryptographic algorithm and the secrecy of the key. A cryptographic algorithm, plus all possible keys and all the protocols that make it work, comprise a cryptosystem.

2.3.3 PROPOSED SYSTEM

Two approaches for data hiding are identified: packet header manipulation and packet sorting. The packet sorting approach is simulated at the network layer which provides feasibility of packet sorting under varying network conditions. While bridging the areas of data hiding, network protocols, and network security, both techniques have a potential for practical data hiding at the transport and network layers.

2.4 Graphing a Network Using Discrete Mathematics

Since a system topology is a physical format, the characteristics of a graph would be explored by using graph theory to define a network and its characteristics. A graph is a simple visual portrayal of this geometric course of action. Graph speculations incorporate numerous standard definitions

and perceive theorems utilized as a part of depicting the structure of a graph and its qualities. Gary Chartrand (1985).

Graph theory has assumed a vital part in processing including yet not restricted to PC graphics, operating system, and data organization and recovery. A later utilization of graph theory and graphs has been in the displaying of PC and some technique for communication between them. (Winfried Grassmann et al., 1996). In a graph portrayal of a PC network, every vertex is a gadget, and each edge signifies a communication medium. Numerous organizations and colleges have at least one local area network (LANs), which are by and large inside one building or a little geological area. As it may be, there is a wide-region network (WANs), whose node, from a topographical outlook, dwell all through numerous areas, states, and even nations. Graphs are imperative when demonstrating these systems to decide and track unwavering quality and productivity.

Graph comprise of nodes, which are linked by edges. The formal meaning of a graph, the mathematical definition, should along these lines depend on V, the arrangement of vertices, and E, the arrangement of edges. Each edge is connected with two nodes, that is, there is a mapping from the edges to the ordered or unordered match of nodes.

> Definition: A graph G = (V, E, F) consist of a finite non empty set V called the set of nodes (points, vertices) of the graph, E is said to be the finite set of edge (lines, arcs) of the graph and F is a function that maps from the set of edges E to a set of structured or unstructured pairs of elements of V. If an edge is mapped to an structured pair, it is called a directed edge; otherwise, it is called an undirected edge. (Eric Weisstein).

Note that the meaning of a graph infers that to each edge of the graph G we can relate an organized or unstructured combine of node of the graph. On the off chance that an edge e ϵ E is along these

lines related with organized pair (u, v) or an unstructured pair {u, v}, where u, v ϵ V, at that point we say that the edge e link or join the nodes u and v. The edge e would then be able to represented by u and v written as e = UV. Any combine of nodes that is connected by an edge in a graph is called nearby or adjacent nodes.Regularly it will be helpful to compose a graph G as (V, E) or straightforward G. when referring to edge or vertices of G. These set can be composed as E(G) and V(G) respectively to recognize the graph name to which the set are related. In the previous case, each edge is specifically represented as the pair that it is mapped to, which wipes out the need to determine f if f is one-to-one mapping.

Example: the graph FIASSEC = (V, E) is a limited undirected graph where V= {Boye, Sele, Awoin, Sidney, Narobi} and E = {{Boye, Sele}, {Sele, Awoin}, {Sele, Sidney}, {Awoin, Sidney}, {Sidney, Narobi}}.The FIASSEC network will be referenced throughout this research as a example to illustrate the concepts and methods proposed herein. The said network is shown in figure 2.1

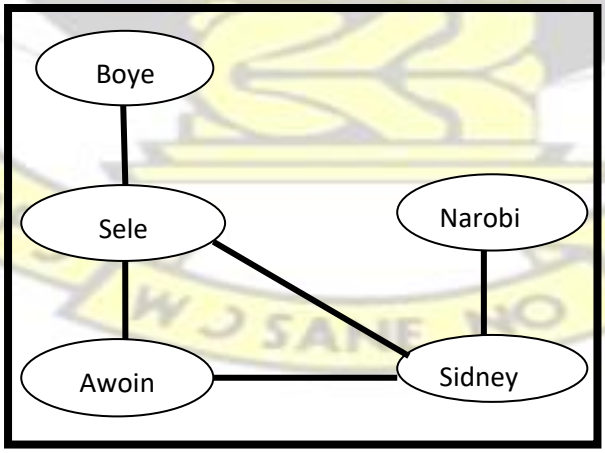


Fig 2.1 Sample of FIASSEC network consisting of 5 node and 5 links as described.

2.4.1 Connected Versus Disconnected

According to Winfried Grassmann et al., 1996, given that G = (V, E). A grouping of the edge is known as a path of G if and just if the terminal node of each edge in the underlying node of the next edge, assuming any, in the path. An undirected graph is stated to be connected if any pair of nodes in the graph are reachable from one another. A graph that is unjoined is said to be separated. The suspicion in this thesis is that we are managing one network at any given moment and that the network is connected.

2.4.2 Labeled and Unlabeled Graphs

A graph with edges or vertices that have been doled out particular qualities or marks is intended as a marked graph. As networks have node names or some kind of node identifier the relating graph that incorporates this data would be thought to be a vertex named graph. On the off chance that the edges likewise have names, at that point, the diagram would be thought to be a vertex and edge named graph. Since edge identifiers are not some portion of the extent of this work, all graph of a network is considered to be vertex marked.

2.4.3 Order and Size

The amount of vertices in G is called the order of G. Whiles the amount of edges in G is named the size of G. Since G = (V, E) where V is the collection of vertices or nodes, and E is the set of edges we can write |V| = order of G and |E| = size of G. (Gary Chartrand, 1985)

Example: In the FIASSEC network |V| = the order of FIASSEC = 5, since there are five nodes: Boye, Sele, Narobi, Sidney, and Awoin. The size of FIASSEC, written as |E| is the number of

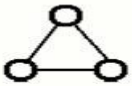
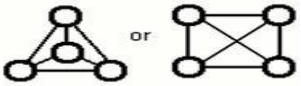
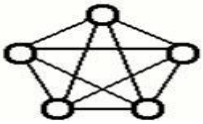
edges in the graph, which is 5: {Boye, Sele}, {Sele, Awoin}, {Sele, Sidney}, {Awoin, Sidney}, and {Sidney, Narobi}.

2.4.4 Degree

The amount of edges of G incident with v where v ϵV(G) is called the degree of v in G. The degree of v is denoted by degG vi. (Gary Chartrand, 1985, Bernard et al. 1996). For an undirected graph G, the sum of the degrees of the vertices of G equals twice the amount of edges of G. Symbolically, if G has order p, which is the number of vertices, and size q, which is the number of edges, with vertices v1, v2, … vp, then ∑ (degG vi) = 2 * q. (Gary Chartrand, 1985). Example: In FIASSEC the degree of the node Sele is 3 since there are 3 edges (3 links) incident with Sele: {Boye, Sele}, {Sele, Awoin}, and {Sele, Sidney}. The degree of each of the nodes are Boye =1, Sele = 3, Awoin = 2, Sidney = 3, and Narobi = 1 respectively. The order of FIASSEC is 5, and the size of FIASSEC is 5. For FIASSEC the ∑ (degG vi) = 1 + 3 + 2 + 3 + 1 = 10 = 2 * q = 2 * 5.

2.4.5 Completeness

A graph G = (V, E) is declared to be complete if every node is adjacent to all other nodes in the graph. In other words, a graph is complete if every distinct pair of vertices in V are adjacent. A complete graph of p nodes is indicated by K. (Gary Chartrand, 1985). Furthermore, if G is a complete graph, then |E| = (p*(p-1)) / 2 where p = |V| as illustrated in figure 1.4.

|V| = p	K_p	$\|E\| = \frac{p*(p-1)}{2}$	Samples Graphs
2	K_2	1	
3	K_3	3	
4	K_4	6	
5	K_5	10	

G1

G2 G3

G4

Fig 2.2

Complete graphs for |V| = 2,…5, the $K_p$notation and corresponding |E|.

For a connected graph the minimum value for |E| is |V| - 1. If $|E| \leq |V| - 2$ then the graph cannot be connected. A graph that has |E| = |V| - 1 is not necessarily connected; but, any fewer number of edges and the graph is guaranteed to be disconnected.

Example: FIASSEC has |V| = 5 and |E| = 5; therefore, FIASSEC is not complete since by the definition of completeness, if |V| = 5, then |E| which equals (p*(p-1)) / 2 must equal 10.

Completeness is easy to check for if you know |V| and |E| for any graph.

2.4.6 Length and Distance

Given G = (V, E). A sequence of edges is called a path of G if and only if the terminal node of each edge in the path is the initial node of the next edge, if any, in the path. The number of edges appearing in the sequence of a path is called the length of the path. In other words, given the path between u and z of sequences of edges {uv, vw, wx, xy, yz} the length of u to z path is the number of edges in the sequence which in this case is 5.

The distance d(u,v) between two vertices u and v of a finite graph is the minimum length of paths connecting them. (Eric Weisstein). It is likely to have multiple paths that exist between a pair of

nodes. The distance between the pair is the shortest path. The distance between a pair of nodes that are adjacent is 1. In network terminology, the length is sometimes regarded as the "hop count."

Example: The path from Boye to Sele to Awoin to Sidney to Narobi is a path in FIASSEC network from Boye to Narobi. Its length is 4. An alternate path would be from Boye to Sele to Sidney to Narobi with a length of 3. The distance between Boye and Narobi is 3, that is, the minimum length of any path from Boye to Narobi. Specifically, the final path of Boye to Sele to Sidney to Narobi is the shortest path as illustrated in figure 2.3.

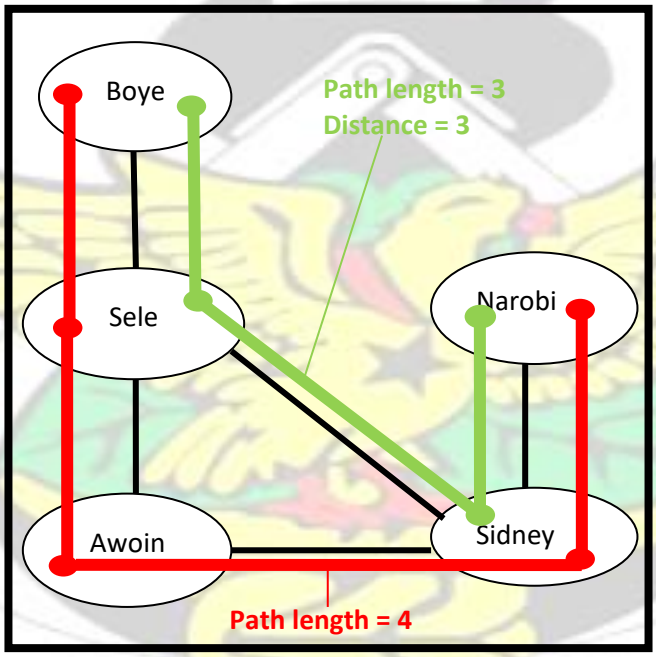


Fig 2.3 Two paths between Boye and Narobi with lengths 3 and 4.

2.4.7 Relationships between the Metrics

Connections exist between the qualities talked about in the past sections. Thus changing one trademark may influence others. In this manner, understanding the interrelationships between these characters will detail how sanitization on the information may affect the outcomes.

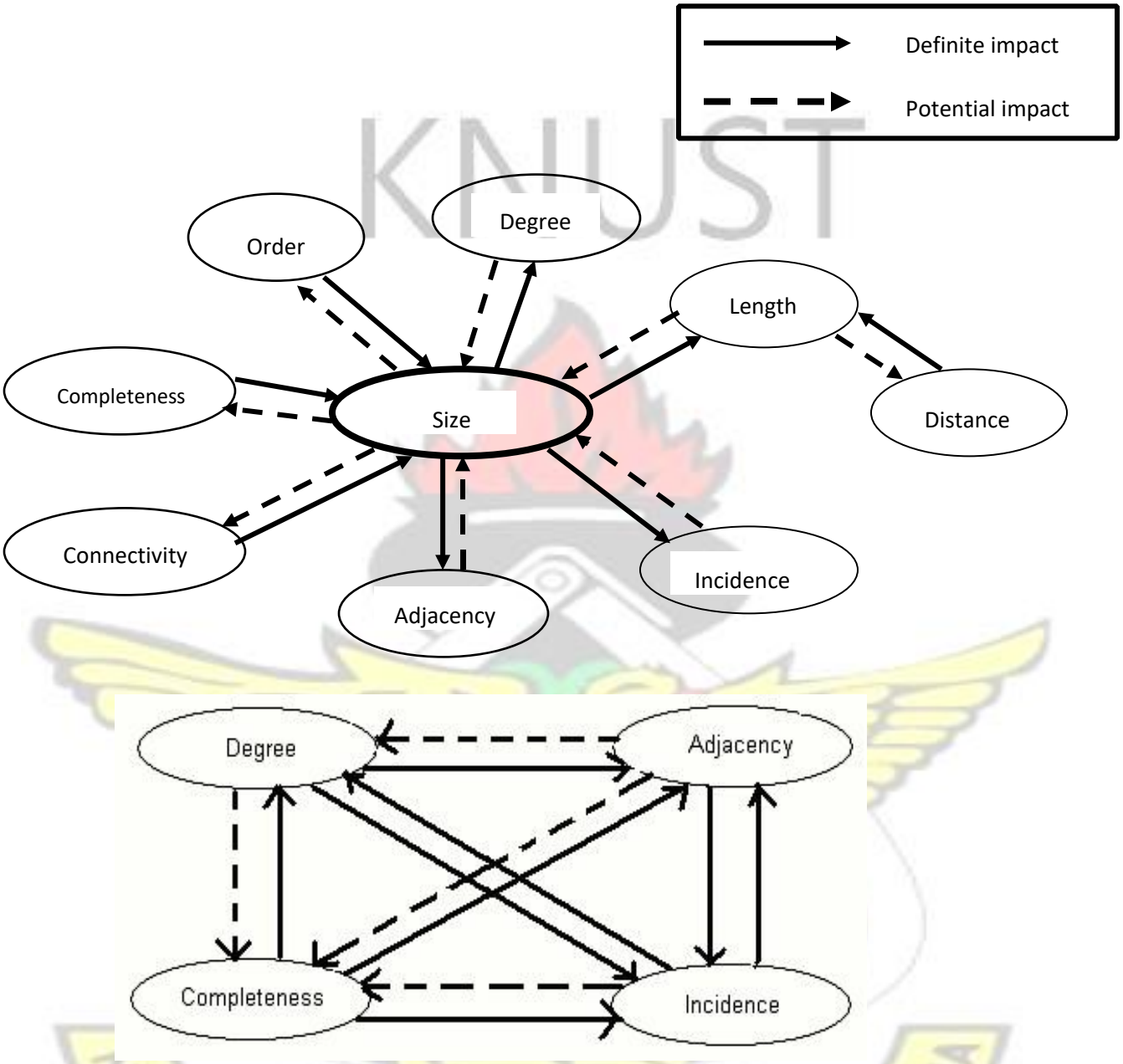


Fig 2.4 Relationships between the different graphs attributes.

The diagram above represents the associations that these attributes have with each other. This diagram centers on the adjustments in the size of a network, that is, the number of connections contained in a network. The supposition being rolled out here is the improvements in the order (i.e., the number of nodes in the system) result in changes in the size, in this manner, looking after availability. At the end of the day, if a node is evacuated or disengaged, at that point all connections

linking it to the graph are additionally expelled. Additionally, while adding a node to the system no less than one edge must be added to link the new node to the network.

CHAPTER THREE

METHODOLOGY

3.0 Introduction

This chapter presents the methodology of the project. The implementation to sanitize a defined network aimed at encapsulating the traffic or data disseminated through the network as well as the characteristics of the network.

It also captures the definition of the said network, producing a syntactical representation of the network using a syntax definition language such as BNF or EBNF, etc. The network sanitization algorithm is applied to the syntax of the defined network to yield the expected result(s) that will necessitate a comparative analysis or study of the outcomes of the sanitization algorithm. This will finally help to establish the relevance of the methodology and the project at large especially concerning the resilience and security of the network.

In the next section, details are provided about the research strategies employed to solve the research issues recognized above, implementations, coupled with the modality of data collection for analyzing the data including laboratory experiments and observation, and then proceed on the conceptual framework for data analysis. Moreover, spiny issues relating to potential limitations and challenges inherent in the selected research strategy and its implementation.

3.1 Research Strategy

The empirical research in this study is interested in proving the hypothesis that, if network data or traffic is well sanitized, there will be a commensurate improvement in the encapsulation of a particular network topology. It will also be used to improve the protection of data disseminated through the network making it difficult for insiders to decipher such data.

This was carried out by setting up a virtual network topology, producing a syntactic representation of the network and the data sanitized to encapsulate the network topology and other relevant data and network information. The determination of a research strategy depends on the characteristics and nature of the research since according to Yen (2003), there are various forms of research strategies which can be; Experiments, survey, case study, etc.

The suitable research strategy selected for such an empirical research is the experimental research strategies.

Experimental research strategies aim at manipulating certain environmental or external conditions and analyzing or examining how those conditions or behavior is affected, and this is done in a highly deliberate and systematic manner. About four main characteristics make experimental research different from other research strategies like the case study. These are;

i) It is under control which implies the removal or minimization of the influence of such variables by different methodologies, such as randomization.

ii) Manipulated where there is intentional manipulation of the indicators or conditions by the research team or researcher.

iii) identifies the manipulative effect(s) of the dependent and independent variables. iv) Replication which also deals with undertaking many sub-experiments instead of one.

These and other characteristics of Experimental Research make it the best Research strategy for this project.

3.2 Research Approach

In essence, experimental research such as this is qualitative but not quantitative. Qualitative research focuses more on collecting, analyzing, and interpreting data using observing what people do or say and it is subjective. In quantitative research, greater emphasis is placed on measurements and quantities. Hence the majority of the scientific research which undertaken tends to be quantitative research because it deals with quantifiable data. This experiment was carried out on an 8GB memory, Intel Core i5 processor, and 500GB hard disk size desktop machine.

3.3 Data Collection: Experiment and Observation

This research is much of qualitative and will be undertaken using experimental research strategy. Data from the experiment will be gathered or retrieved to perform in-depth analysis to prove the hypothesis.

Observation (which involves looking carefully) is another technique used in collecting data in this research. Observation includes looking carefully. Overcome the challenges of observation as a data collection technique, self- inference, and effect were fully minimized during the testing process. When observation is used to collect data, the researcher examines and investigates the impact of the modification of the dependent and independent variables.

Therefore, by the above definition, the suitable methodologies to be employed in undertaking this research are by Experimentation and Observation. Experiments were conducted out by setting up a virtual desktop environment with Windows XP, Windows 7, as well as vulnerability and penetrating test system software called BackTrack 5, were installed on the virtual platform to undertake network vulnerability and penetrating testing to ensure the resilience and robustness of

the network. Here, at each stage of the experimental process, careful observations were made to examine the process at each stage to gather or retrieve the data needed to undertake this project.

3.4 Penetration Testing

3.4.1 Setting up a Penetration Platform

This platform enables us to undertake a series of penetration test which includes vulnerability assessment and network mapping.

First and foremost, a virtual platform was setup by installing VMware on the host computer. On this virtual platform, different Operating Systems such as Windows XP and Windows 7 was installed on it to serve as the guest systems on the host computer.

Also, Backtrack 5, hacking software which contains almost all the hacking tools needed for a penetration test was also installed on the virtual platform.

All the machines, i.e., the host as well as the virtual guest computers were networked and were communicating with each other as in real life situation.

After all these setups, the system was ready for the penetration test to be done. The penetration and vulnerability testing were performed on the initial system. After enhancing the network with sanitization and some countermeasures, the previously confirmed penetration test was again carried out on the network. Subsequent chapters explain further how the testing was done and their equivalent results respectively.

3.4.2 Penetration Testing on Initial Network

Scanning Pen Testing: here a step by step procedure penetration test was conducted on the target network following the flowchart as shown in figure 3.1.

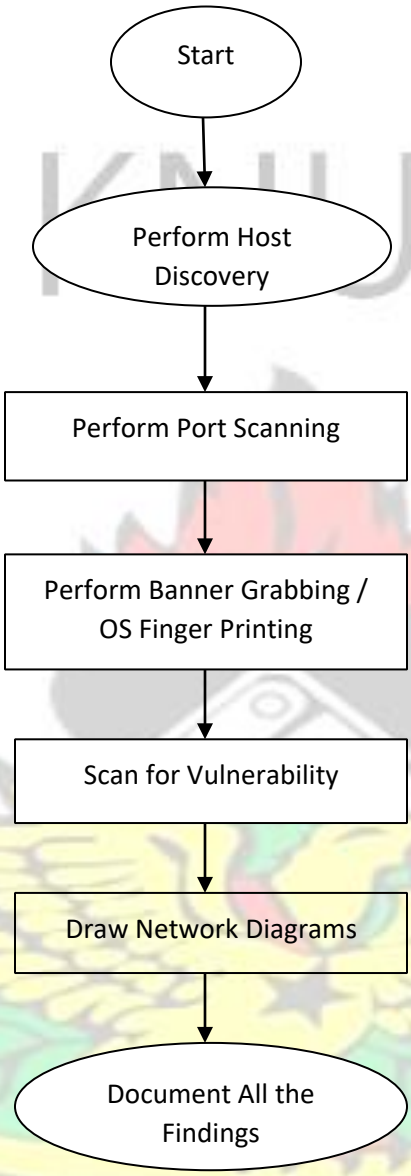


Fig. 3.1 A Flowchart Showing Step by Step procedure for the penetration test.

Step 1: Host Discovery

The host discovery testing was performed to discover live host on the target network via a network scanning tool called Nmap included or embedded in the Backtrack 5 software. The Nmap tool was used to ping scans command, i.e., a targeted guest computers IP address was entered into the Nmap tool, as indicated in the Target Box in figure 3.2 and its corresponding command as shown in the

Command Box also in figure 3.2. And finally, the scan button was clicked for an ICMP ECHO requests to be sent from the virtual host to the guest on the network. The live hosts responded to the ICMP ECHO requests with an ICMP ECHO reply. Hence, the host discovery scan was useful for locating active devices or determining if the request passed through a firewall. Figure 3.2 provides a snapshot of the host's discovery scan command and the respective outcome.

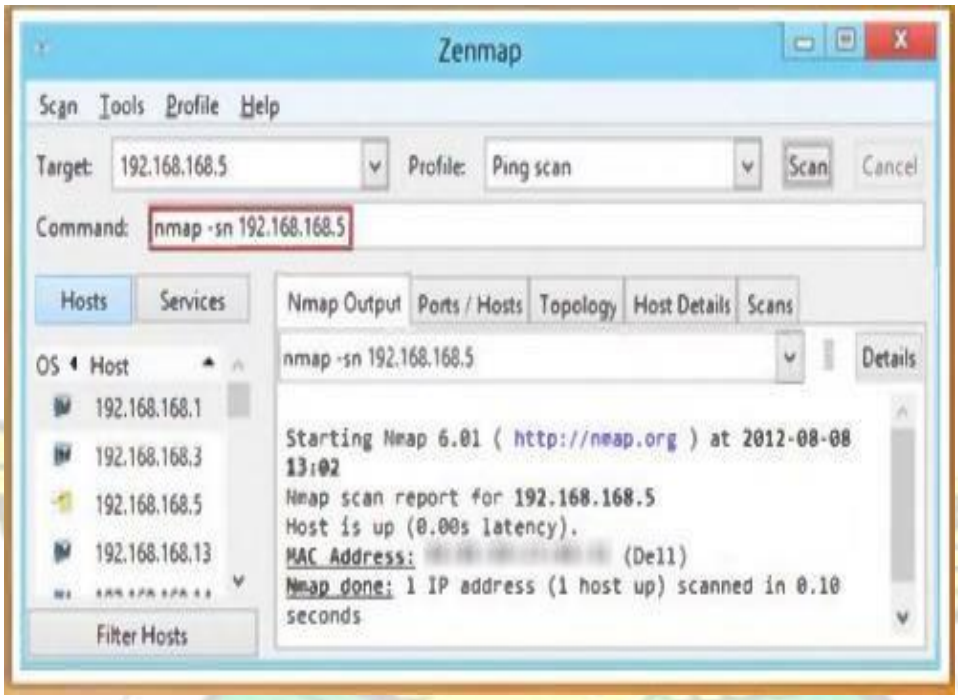


Fig. 3.2. A Zenmap screenshot showing Ping scan output

Step 2: Port Scanning

After the host discovery scanning test, a port scanning test was undertaken using the Nmap tool. The intent for the port scanning was to check or determine whether a server or host on the target network has an open port. Thus, testing to find out the susceptibility of the host to possible attack since the open ports serves as doorways for attackers to install malware on a particular system.

The Nmap tool embedded in the Backtrack 5 software was used to ping scans command, i.e., a targeted guest computers IP address was entered into the Nmap tool (Zenmap) as indicated in the

Target Box in figure 3.3, and its corresponding command as shown in the Command Box also in figure 3.3. And finally, the scan button was clicked for an ICMP ECHO requests to sent from the virtual host to the guest on the network. The live hosts responded to the ICMP ECHO request with an ICMP ECHO reply. Hence the Port Scanning was useful for locating all devices with an open port. This activity and its result illustrated in the figure3.3.

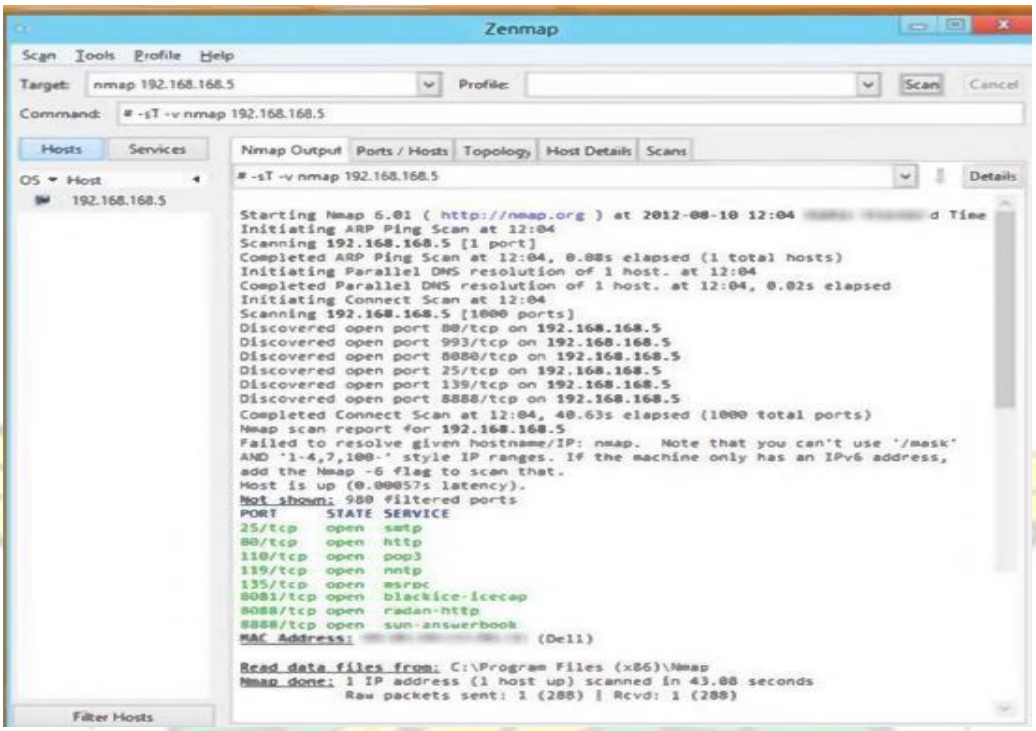


Fig. 3.3. A Zenmap screenshot showing Port scan results

Step 3: Banner Grabbing / O.S Finger Printing

This network penetrating test activity was them performed to ascertain the operating system running on the target host or client of a network as well as the version of the operating system. This activity was successfully carried out using Telnet and the Netcrafttool which is embedded in the Backtrack 5 software by using a ping scans commandie a targeted guest computers IP address was entered into the Netcraft tool as indicated in the IP address Box in figure 3.4 (A) and it corresponding command as shown in the Command prompt also in figure 3.4 (B) and finally the

scan button or the enter key was clicked for an ICMP ECHO requests to be sent from the virtual host to the guest on the network. The live host responded to the ICMP ECHO request with an ICMP ECHO reply by showing all operating systems running on the hosts with their details as shown in figure 3.4. Every operating system has some inherent vulnerability associated with them. Hence, the banner grabbing activity was undertaken to help preempt possible vulnerabilities that potential attackers can harness to gain control over the system and compromise the whole network. The outcome of the banner grabbing / OS finger printing is shown in the figure 3.4 in the red border lines around it.

Fig. 3.4. (A) A Netcraft screenshot showing a Banner Grabbing scan result.

Fig. 3.4. (B) A Netcraft screenshot showing a Banner Grabbing scan result.

Step 4: Vulnerabilities Scanning

This is done to determine the weakness or loopholes of the target system or network. Such vulnerabilities tested included the software update status, firewall-related issues, malware protection issues, unauthorized applications, user credential setup, service packs and updates as well as other related vulnerabilities. This activity was undertaken using a network vulnerability scanning tool known as GFI LANGuard embedded in the Backtrack 5 software by using a ping scan command by sending ICMP ECHO request to the host on the network. The host then responded to the ICMP ECHO request with an ICMP ECHO reply. By showing the following details of the scanned host that's the software update status, firewall-related issues, malware protection issues as well as other related issues as shown in the figure 3.5 with the caption Security Services with the color coding showing Red as high risk, Yellow as medium risk, Green as low risk and Gray as not available.

Fig. 3.5. A GFI LanGuard screenshot showing Vulnerabilities scan result.

Step 5: Draw Network Diagram

After the vulnerabilities scanning test, namely the software update status, firewall-related issues, malware protection issues, unauthorized applications, user credential setup, service packs and updates, a draw network diagram scanning test was undertaken using the LAN Surveyor also a tool embedded in the Backtrack 5 software. The intent for the draw network diagram scanning was to check or determine the critical path on the target network as well as to identify all types of devices such as switches, hubs, firewalls on the network which attackers can stay on to compromise the system. In achieving this, a ping scan command sent across the target network. Here all the live host on the network responded to the ICMP ECHO request with an ICMP ECHO reply by showing the network diagram (the positions and names of the hosts and devices on the network). This activity and its result are illustrated in figure 3.6.

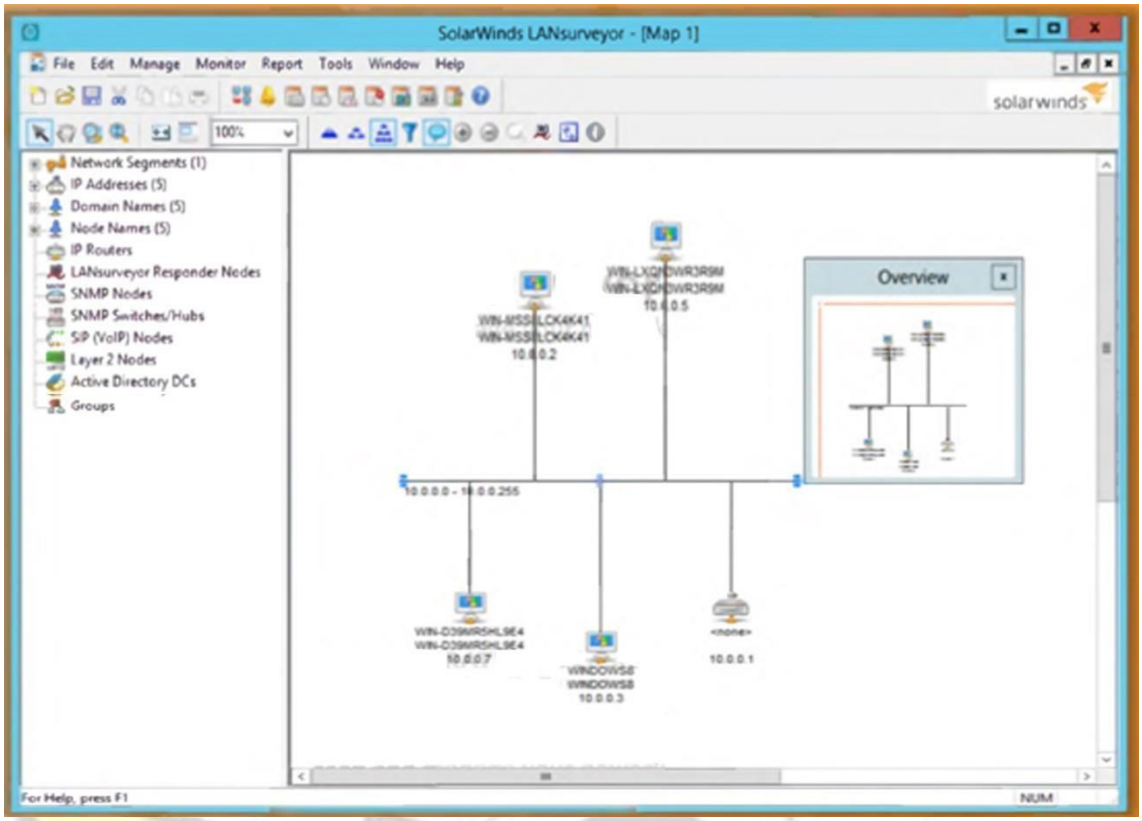


Fig. 3.6. A LANsurveyor screenshot showing Network Diagram result

Step 6: Document All Findings

The findings of the penetrating testing activities were well documented as shown in table 3.1 to serve as baseline measurement criteria to test for any enhancement performed on the network to improve its susceptibility levels. Thus, the documented findings provide a primary and effective source for providing useful counter measures to secure the network and also to compare the outcomes after implementing the counter measures.

TEST	FINDINGS

1.Host Discovery	It was found out that; the system was having some live host which the attackers can use compromise the system or network.
2.Port Scanning	Here it was found out that, some port were opened which can serve as a door way for attackers to install their malware into the system.
3.Banner Grabbing/O.S. finger printer	This test displayed all the operating systems running on the entire computer on the network, which helps hackers to exploit the vulnerabilities specific to that operating system.
4.Vulnerabilities Scanning	It was found out that the system was of high risk and can be easily attacked by hackers.
5.Draw Network Diagram	It was found out that, the network diagram which was the outcome of this test can help hackers to know the critical path on the target network and also determine the types of devices on the network which attackers can stay or to compromise the system.

Table 3.1 Documented findings for the penetrating test

3.5 Standard Syntax for Network Sanitization

Visualizing and representing a network as a graph is simple and straightforward, a graph may not be the most proficient approach to store and offer this data. Therefore, there is the need to get a strategy to convey the network data standardized that is more great for recording, putting away, sharing and sanitizing this data. Achieve this, a standard linguistic structure (syntax) utilized so any system can represent in this linguistic structure. To achieve this, the BNF and Extended BNF (EBNF) for XML a standard syntax were utilized to speak to any system in this language structure. BNF is an acronym for "Backus Naur Form." BNF is documentation that is used to determine the structure of a substance, changing from the sentence structure rules for a programming dialect, to produce orders and in addition to convey protocols. In any case, for this situation, BNF is utilized to portray the topology of the network, to make a different linguistic structure (syntax) for every particular kind of network topology to make managing each type brief and also keep it syntax

essential as would be prudent. Having a precise definition is basic to comprehend the meta-symbols and the general BNF linguistic structure (syntax) to maintain a strategic distance from any error of the grammar. The meta-symbols are unique images used to characterize the semantic structure rules. Guidelines inside this language structure described utilizing the accompanying arrangement appeared in the table3.2.

SYMBOL	EXPRESSION
::=	is defined as
| choice	Or
*	Repetitive items, zero or more times
+	Repetitive items, one or more times
?	Optional items
(…)	Grouping
"quoted"	Text terminals
NORMAL	Non-terminals
/*…*/	comments

Table 3.2 Special symbols used to define the syntax rules

In perspective of this, the EBNF will be utilized as the sentence structure (syntax) to portray the network topologies.

For a mesh (additionally known to as a point-to-point) topology, the EBNF is as per the following:

3.5.1 EBNF Code Interpretation

Line No Rule

1 NETWORK::= 'mesh' ('topology')? Network_name '{' M_SPEC '}'

2 M_SPEC::=(M_NODE_ATTR | M_LINK_ATTR)* M_NODE_SPEC (','M_NODE_SPEC)+

3 M_NODE_SPEC ::= 'node' node_name '{' (M_NODE_ATTR)* M_LINK_ATTR) '}'

4 M_NODE_ATTR ::= node _attribute '=' node_value (',' M_NODE_ATTR)*

5 M_LINK_SPEC ::= 'link to' node_name ('{'M_LINK_ATTR'}')* (',' M_LINK_SPEC)*

6 M_LINK_ATTR ::= link_attribute '=' link_value (',' M_LINK_ATTR)*

Line 1 characterizes a network as a 'mesh' utilizing this terminal. The terminal 'topology' is discretionary as it can be considered out of work and in this way pointless. The network_name speaks to the real name of the system took after by the particulars of the mesh topology.

Line 2 records the node and connection properties. Both can be rehashed at least zero times. The credits can be particular to the network to which this sentence structure portrays and are client characterized. Characteristics characterized here are worldwide qualities connected to the whole network and are like this the default value for every node or connection inside the network. Next is the node determination took after by no less than one more hub particular. The most inconsequential Mesh organize that this grammar can portray two nodes associated by one connection. Anything under two node and one connection are not thought to be a Mesh topology as depicted by this linguistic structure.

Line 3 portrays the node as characterized as 'node' terminal took after by the actual name of the node. After that, any nodes characteristics indicated. This can be rehashed at least zero times. In conclusion, is the connection particular which must happen once since each node must associate in a mesh system to no less than one other node. This is the starting node. The node that this is

connected to is characterized in line 5. The node_name parameters resolved from the node names recorded in 5.

Line 4 characterizes a node credit to measure up to particular node esteem. Node properties are constantly optional; however, if recorded, there can be more than one.

Line 5 characterizes the connection determination as a terminal 'connection to' trailed by the real node name where the connection interfaces with. Connection characteristics are recorded straightaway if there are any. This is hunted by any extra connections that are available from the node indicated in line 3. This detail can be rehashed the same number of times as vital. Links are certainly bi-directional.

Line 6 characterizes the connection quality which breaks even with particular esteem. Connection characteristics are constantly optional, however, is recorded, there can be more than one.

Any mesh network can be reliably portrayed utilizing this language. Having this standard language structure (syntax) will make sanitizing the information less demanding. By describing a network in this way, from any one node, the node is just mindful of its quick associations with nearby nodes and not too different nodes inside the network. The graph edges are bi-directional.

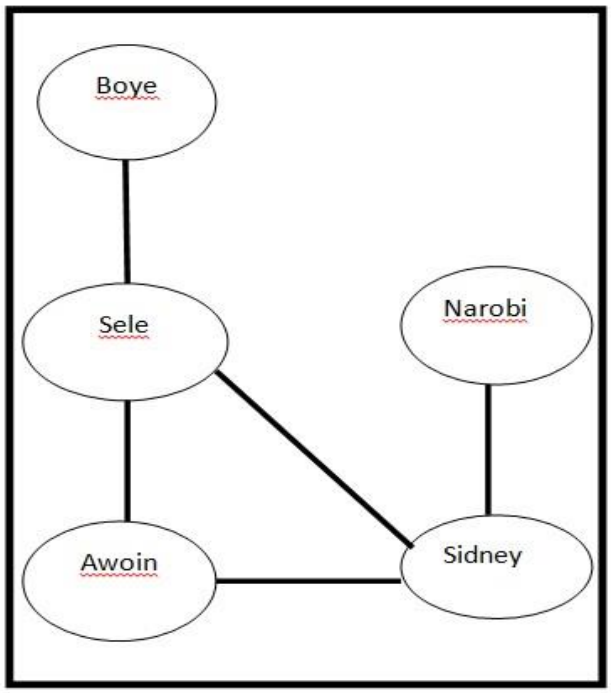
To express the bi-directionality in the language structure, each edge e ϵ E are recorded twice: as (u, v) and (v, u).In relating the linguistic structure (syntax) to the graph parameters, the node_name recorded on line 4 of the sentence structure (syntax) resolved from the node names recorded in V for the graph of the network. The aggregate number of line 4 entries measures up to |V|.

The link_to node on line 6 of the linguistic structure (syntax) likewise recorded in V. line 6 will list all nodes that are nearby the node marked on line 4 on the sentence structure (syntax). The

cumulative number of line 6 routes or entries in the syntax measures up to 2*|E| since each edge is spoken to as a connection twice: uv and vu.

3.5.2 EBNF Syntax applied to FIASSEC network

Given that the FIASSEC network has five nodes and edges, the application of the syntax to the FIASSEC graph results in the syntax having five entries for line 4 and ten total entries for line 6 as illustrated in figure 3.7.




```
Mesh network FIASSEC{
        Node Boye{
                Link to Sele
        },
        Node Sele{
                Link to Boye
                Link to Awoin
                Link to Sidney
        },
        Node Awoin{
                Link to Sele
                Link to Sidney
        },
        Node Sidney{
                Link to Sele
                Link to Awoin
                Link to Narobi
        },
        Node Narobi{
                Link to Sidney
        },
}
```

Fig 3.7 Graph of the FIASSEC network and the corresponding network syntax.

3.6 Uninterrupted and Secondary Network Characteristics

For sanitization purposes for the topology of the system. Exactly, accentuation is on the mesh networks, the node names and the connections between the nodes. As expressed in the writing audit (literature review), there are positive qualities that can be correctly decided easily by watching the syntax. To be specific, the network name, node (names), order (|V|), size (|E|), adjacency, incidence, and the degree of the nodes. They are attributes that will disinfect with a specific end goal to conceal a few or these features of the system. The optional qualities are

completeness checking, paths lengths, distance, and the graph diameter. These qualities of a system require some more inside and out calculations and can be affected relying upon the strategy used to sanitized the immediate attributes. By effectively covering up or veiling the next parts of the system, we plan to cover the indirect ones as well.

3.7 Application of Sanitization Methods to Network Information

The sanitization options to be defined or discussed, aside from deletion, except that the foe or an outsider can access the more significant part of the post sanitization information. This is the most distrustful scenario information availability situation and is a significant pointer to test if the alternatives for sanitization are suitable. The objective is to secure the protection of the information to keep the enemy from recreating the first data from the sterilized data while keeping up similar investigation activities, that is, the consequence of examination on the sanitized information have the same outcomes as if the tasks were executed on the raw data. If there should arise an occurrence of this exploration, we are sanitizing node and connection names to shroud the connectivity between the nodes to conceal the system's topology from a foe while at the same time attempting to keep up the statistical outcome of the data. The foe may have outside information that would make a viable sanitization technique pointless. Likewise, remarkable attributes of a network may accidentally be helpful to the foe as he endeavors to recreate the network. These constraints on the adequacy of the sanitization technique will be examined.

3.8 Fixed Sanitization Applied to Network Syntax

The initial technique for sanitizing is fixed, that is, the system name and node names recorded in the syntax structure are succeeded with a similar optional value. Every event of given vertex name, regardless of whether the name shows up in the node _ name field or in the link _ to handle, will succeed with a similar interchange value. Any fixed string can be utilized for the substitution.

Therefore the kind of the supplanting object requires not coordinate the sort of the object that it is succeeding. The sanitized network data in the syntax makes a graph that is isomorphic to the first graph. The order and size of the purified graph are alike to the unsanitized one. The degree of every node additionally stays unaltered. Besides, the connections between the nodes and edges, that is the nearness and frequency are unaltered. This strategy keeps the path lengths, distance, and graph distance across (diameter) unaltered since the sanitized diagram is alike, aside from the node names, to the first graph. This is a worldwide name substitution plot. By supplanting these qualities in the syntax, we have successfully hidden the first network name and network node names; in this way, securing the protection of this data. In spite of the fact that this technique is not difficult to implement, but there are disadvantages regarding security. With access to the majority of the sterilized information regardless of how vast this graph is, both in order and in size, the main security this strategy has accomplished is in ensuring the network name and node names. After purification (sanitization), the link between the nodes is still clear. By renouncing security of alternate qualities in the system, we hold a same investigation of the cleaned network as well as we would have with the unsanitized one. In the event that the investigator or other outside substance has outer information and knows the names of the system and its nodes, at that point the main staying doubtful is the dimensions used to cover the node names.

Example: Applying the fixed method of sanitization to the experimented network christened“FIASSEC network,” utilizing a straightforward swap for the network and node names from the first identifier to a character identifier, the system syntax would be changed as shown underneath.

In figure 3.8 (A), the length, shape and locating of the edges are not fixed when studying a graph. To effectively imagine that the two graphs are isomorphic, a similar design was utilized. From this

graph appeared in figure 3.8 (A & B) we can see that they are isomorphic. Amazingly, there is just a single conceivable graph that is developed from the sanitized data. Since only a single isomorphic sanitization graph exists, with access to the majority of the post-cleansing information, an enemy has the correct format of the network despite the fact that the network and node names are covered.

{FIASSEC, Boye, Sele, Awoin, Sidney, Narobi} => (Y, D, W, B, J,)

Original syntax

```
Mesh network FIASSEC {
        nodeBoye {
                Link to Sele
        },
        nodeSele {
                Link to Boye,
                Link to Awoin,
                Link to Sidney
        },
        nodeAwoin {
                Link to Sele,
                Link to Sidney,
        },
        nodeSidney {
                Link to Sele,
                Link to Awoin,
                Link to Narobi
        },
        nodeNarobi {
                Link to Sidney
        }
}
```

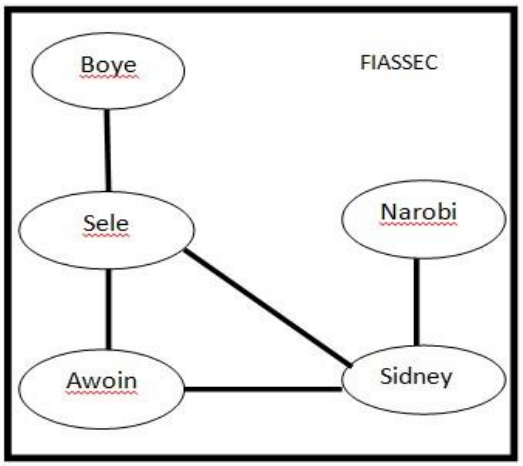


Figure 3.8. (A) Original FIASSEC network syntax and the Graph.

Sanitized (fixedmethod) Syntax

```
Mesh network Y {
        node D {
                Link to W
        },
        node W {
                Link to D,
                Link to B,
                Link to J
        },
        node B {
                Link to W,
                Link to J,
        },
        node J {
                Link to W,
                Link to B,
                Link to H,
        },
        node H {
                Link to J,
        }
}
```

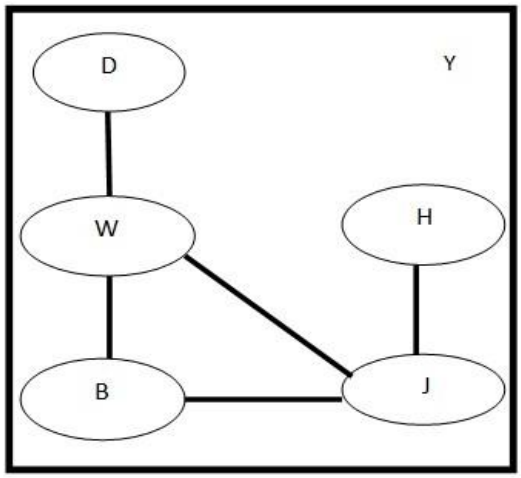


Figure 3.8.(B) Sanitized FIASSEC network syntax and Graph, called Y, using fixed method.

3.8.1 Fixed Sanitization Applied to A Complex Network (RedIRIS)

RedIRIS, the national research, and education network (NREN) for Spain or the national spine network of Spain and gives access to colleges and research focuses inside this nation. The network comprised of 18 regional nodes and focal node interconnected by a mesh organizes.

The order of this system is 19; the size is 31 as illustrated in figure 3.9

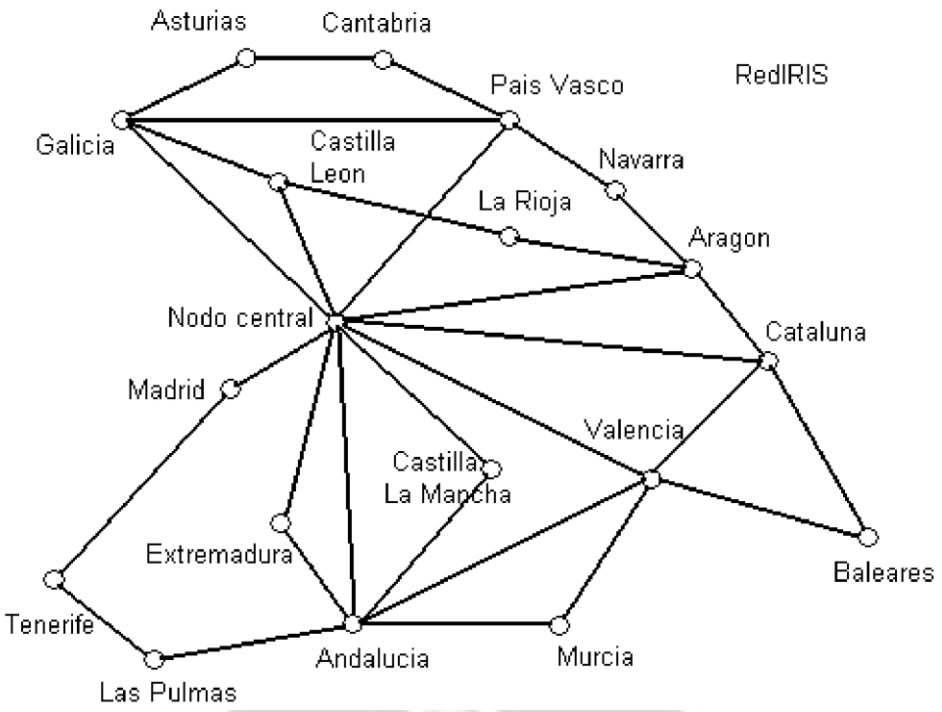


Figure 3.9 Graph of the RedIRIS network

These are the list of the nodes {Galicia, Asturias, Cantabria, Pais Vasco, Navarra, Aragon, Cataluna, Baleares, Castilla Leon, La Rioja, Valencia, Murcia, Andulucia, Las Palmas, Tenerife, Madrid, Extremadura, Castilla La Mancha, Nodo Central} and the resulting values { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19 } to be used in replacing the nodes for the sanitization along with the corresponding sanitized network syntax.

```
mesh network RedF {
        node 1 {
                link to 2,
                link to 4,
                link to 9,
                link to 19
                },
        node 2 {
                link to 1,
                link to 3
                },
        node 3 {
                link to 2,
                link to 4
```

```
},
node 4 {
        link to 1, link
        to 3, link to
        19, link to 5
        },
    node 5 {
        link to 4, link
        to 6
        },
    node 6 {
        link to 5, link
        to 10, link to
        19, link to 7
        },
    node 7 {
        link to 6, link
        to 11, link to
        19, link to 8
        },
    node 8 {
        link to 7,
        link to 11 },
    node 9 {
        link to 1,
        link to 10,
        link to 19
        },
    node 10 {
        link to 9,
        link to 6
        },
    node 11 {
        link to 7,
        link to 8,
        link to 19,
        link to 12,
        link to 13
        },
    node 12 {
        link to 11,
        link to 13
        },
    node 13 {
```

```
link to 12, link
to 11,
        link to 19, link
        to 18, link to
        17, link to 14
        },
node 14 {
        link to 15, link
        to 13
        },
node 15 {
        link to 14, link
        to 16
        },
node 16 {
        link to 15, link
        to 19
        },
node 17 {
        link to 13, link
        to 19
        },
node 18 {
        link to 13,
        link to 19
        },
node 19 {
        link to 1,
        link to 9,
        link to 4,
        link to 6,
        link to 7,
        link to 11,
        link to 18,
        link to 13,
        link to 17,
        link to 16
        }
}
```

The subsequent graph is a one of a kind isomorphic diagram to the firstRedIRIS organize network. The order and size of the diagram are indistinguishable from the RedIRIS graph appeared in figure 3.10

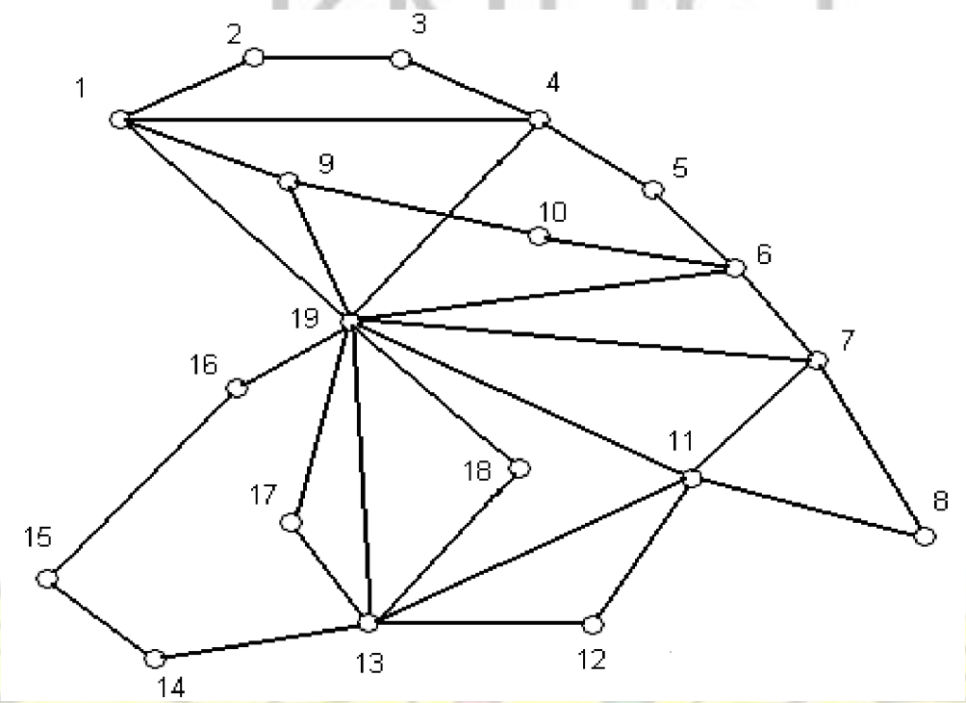


Figure 3.10 Resulting graph from fixed transformation of RedIRIS network.

3.9 Variable Sanitization Applied to Network Syntax

Variable sanitization takes into consideration the context and structure of the information that's to be changed. This means that, information representing one form of the item can have one mapping whereas another object incorporates a separate mapping. This distinction will cause a resourceful worth to be mapped to two completely different values with sanitizations counting on the context within which the worth happens. This technique is a lot complicated than fixed sanitization and needs rules to outline that fields to be mapped employing a specific operation. Within the following sections two completely different variables sanitization schemes square measure mentioned here namely:

- Variable Sanitization with separate Mappings
- Variable Sanitization with Unique Names.

3.9.1 Variable Sanitization with separate Mappings

This variable of network language structure sanitization has diverse mapping capacity for the link_ to objects and the node_ name objects regardless of whether the estimation of these two objects is the same. For instance, in medicinal records, If purifying people first name and the last names, there can be distinctive mappings utilized for each field. In this manner, If "Clark" showed up as the last name, it may be mapped to "Jones" though if "Clark" showed up as the primary name of a person, it could be mapped to "Brad."

With this kind of sanitization, we are not influencing the order or size of the system since you are just changing the node _name and link _ to node names. Moreover, since only the names are changing, the degree for every node, adjacency, and incidence will be as before. So the security of the names of the nodes is ensured similarly likewise with the fixed transformation. Also, included the level of complex nature exists with the variable strategy, however since there will be two mapping capacities for the system nodes: one for the node_ name field and one for the link _ to the field. One of the field impacts of this change is that, due to the two capacity mapping, it will be harder (however not unimaginable) for the substance or foe examining the information to decide the graph of the system given the sanitized system syntax. This additional measure of protection to the system might be endangered if the foe can recreate the network graph from the system syntax. On the off chance that reproduction of the graph is accomplished, at that point the indirect system qualities, for example, path length, distance and the graph distance across (diameter) can be resolved. On the off chance that the foe cannot remake the diagram, at that point the protection

of the data is maintained. Thus, this data could be assessed, however not known completely since part or the majority of the data is uncertain.

Illustration: Applying the variable sanitization technique to the FIASSEC network, utilizing a basic trade for the node_ name from the first identifier to a character identifier. And a separate substitution work for the link _ to names from the first label to a character string, the system sentence structure (syntax) would be changed, and the outcoming graph is shown in figure 3.11 (B).

Network_name: {FIASSEC} => {Jeff}

Node_name: {Boye, Sele, Awoin, Sidney, Narobi} => {D,W,B,J,H}

Link_to: {Boye, Sele, Awoin, Sidney, Narobi} => {Toyota, Kia, Opel, Ford, Benz}

Original syntax

```
Mesh network FIASSEC {
        nodeBoye {
                Link to Sele
        },
        nodeSele {
                Link to Boye,
                Link to Awoin,
                Link to Sidney
        },
        nodeAwoin {
                Link to Sele,
                Link to Sidney,
        },
        nodeSidney {
                Link to Sele,
                Link to Awoin,
                Link to Narobi
        },
        nodeNarobi {
                Link to Sidney
        }
}
```

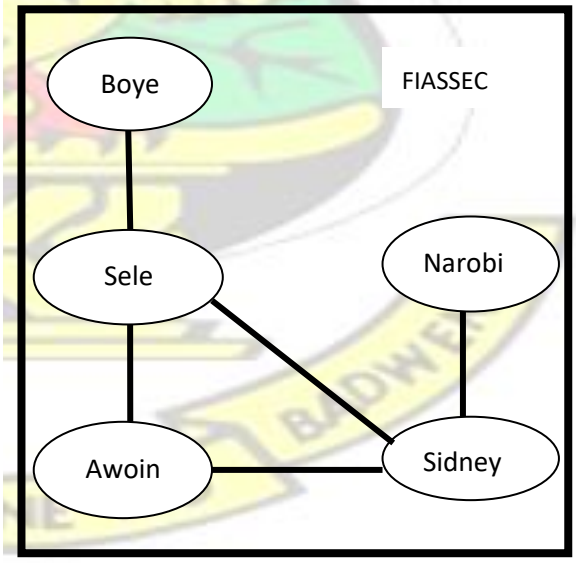


Figure 3.11 (A) Original FIASSEC network syntax and the Graph.

Sanitized (flexible method) Syntax

```
Mesh network Jeff {
        node D {
                Link to KIA
        },
        node W {
                Link to TOYOTA,
                Link to OPEL,
                Link to FORD
        },
        node B {
                Link to KIA,
                Link to FORD,
        },
        node J {
                Link to KIA,
                Link to OPEL,
                Link to BENZ,
        },
        node H {
                Link to FORD,
        }
}
```

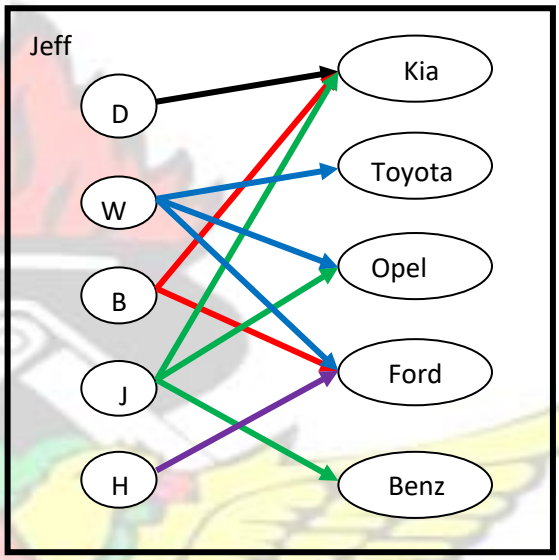


Figure 3.11 (B) Sanitized FIASSEC network syntax and Graph, called Jeff, using variable sanitization.

The more nodes in the graph, the harder the remaking can be. By looking at each arrangement of nodes, the node_ name and link_ name to sets, the degrees inside each set to compare to the first graph. That is, in the first graph there are five nodes of degrees 1,1,2,3 and 3. For each arrangement of nodes in the sterilized Jeff graph, there are five nodes of degrees 1,1,2,3 and 3.The order and size of the Jeff graph is twice that of the FIASSEC graph because of utilization of two mapping capacities used to sanitized the nodes. Since in the Jeff graph there is just a single node in every node mapping set with a degree of 2, an enemy can find out that node B and node Opel are similar nodes.

Moreover, since W and J alongside Kia and Ford all have a degree of 3, and since a node cannot have an edge to itself, at that point W can't be Ford and J can't be Kia. In this way, W must be Kia and J must be Ford. The chief doubtful would be which node Toyota and Benz are: D or H. Accordingly there are two changes of the reproduced graph. The more significant part of the connections in a changed graph is bi-directional as appeared in figure 3.12.

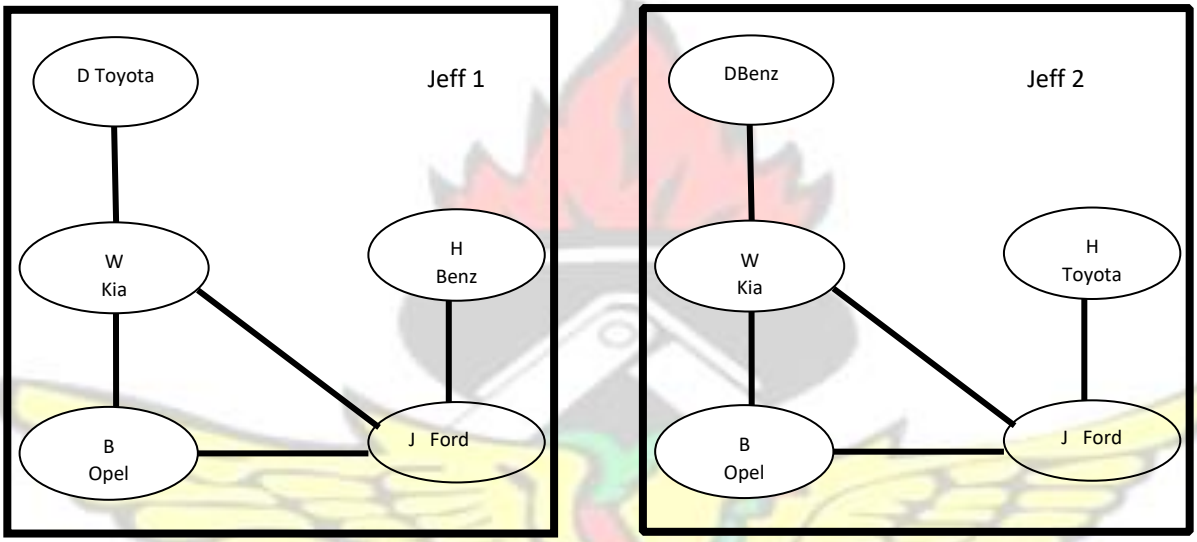


Figure 3.12. Two transformations of the Jeff network

3.9.2 Variable Sanitization with Unique Names

The other is the variable sterilization of the link_ to nodes names in light of their relationship to the nodes _ name. Toward the day's end, each link_ to a node with a similar contrasting node_ name will be mapped with the same regard. Besides, since in the language structure (syntax) we don't think about more than one association between a few nodes, no link_ to a node will have the same node_ name so each link_ to a node will be mapped to an exceptional value. Using the first and last case in restorative records, If "Smith" is the last name, it would be mapped to "Johnson." If "Clark" appeared as the central name with "Smith" as the last name, by then "Clark" would be mapped to "Brad." If the last name was not "Smith," by then "Clark" as the essential name would be mapped to some other regard, say "James." Hence, "Clark Smith" would have a

novel mapping to "Brad Johnson" however "Clark Jones" would be mapped to "James Brad" (tolerating that "Jones" mapped to "Brad"). Along these lines, there would not be where an enemy could distinguish similitudes between the main names since the capacity mapping for the principal names is reliant on the last names.

With this sanitization technique there are two particular mapping capacities, and the link_ to nodes names are interestingly mapped, the interval result will be a disengaged directional graph with order three names times as extensive, and the size doubles the huge as the first graph. This sanitization technique is more mind confusing than the past two strategies because of the one of a kind mapping highlight and multifaceted nature associated with changing this graph into a connected bidirectional graph. Similarly, as with sanitization utilizing separate mappings, It will be hard to remake this bidirectional graph. As the quantity of nodes expands so will the recreation difficulty. If remaking is accomplished, at that point the direct and indirect network qualities can be computed. With graphs that are minor or complete, this strategy will not be successful. Example: Applying the variable sanitization strategy to the FIASSEC network, utilizing the one of a kind naming replacement for the link_ to node names from the first identifier to a numerical identifier in view of which the node _ name. The outcome is a unique mapping of the considerable number of nodes.

Network _ name: {FIASSEC} -> {Royal}

Node _ name: {Boye, Sele, Awoin, Sidney Narobi} -> {D. W, B, J, H}

Link_to: { Boye (node Sele), Sele (node Boye), Sele (node Awoin), Sele (node Sidney), Awoin (node Sele), Awoin (node Sidney), Sidney (node Sele), Sidney (node Awoin), Sidney (node Narobi), Narobi (node Sidney)} -> {1,2,3,4,5,6,7,8,9,10}

Original syntax

```
Mesh network FIASSEC {
        nodeBoye {
                Link to Sele
        },
        nodeSele {
                Link to Boye,
                Link to Awoin,
                Link to Sidney
        },
        nodeAwoin {
                Link to Sele,
                Link to Sidney,
        },
        nodeSidney {
                Link to Sele,
                Link to Awoin,
                Link to Narobi
        },
        nodeNarobi {
                Link to Sidney
        }
}
```

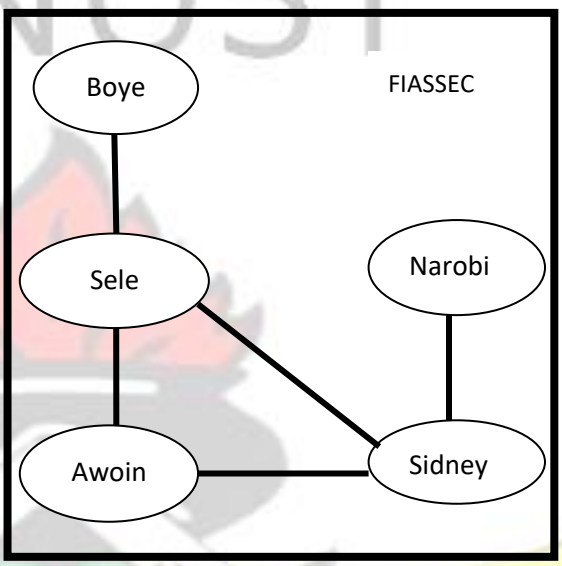


Figure 3.13 (A) Original FIASSECnetwork syntax and Graph.

Things remain the same where the length, shape and situating of the edges are not settled when interpreting a graph. The test an enemy has is to recreate the sanitized information as a bidirectional associated graph. The greater the size of the graph that is, the bigger the quantity of nodes, the harder the task to recreate a bi-directional graph.

By looking at the left arrangement of nodes in the Royal graph, the degree's inside this set do compare to the first graph. That is, in the first graph there are five nodes of degrees 1,1,2,3 and 3. For each arrangement of nodes in the sanitized Royal graph, there are five nodes of degrees 1,1,2,3 and 3. The order of the Royal diagram is three times bigger, yet the size is still, likewise with the past variable technique, twice that of FIASSEC as appeared in figure 3.13 (B).

Sanitized (variable method) Syntax

```
Mesh network Jeff {
        node D {
                Link to 2
        },
        node W {
                Link to 1,
                Link to 5,
                Link to 7
        },
        node B {
                Link to 3,
                Link to 8,
        },
        node J {
                Link to 4,
                Link to 6,
                Link to 10,
        },
        node H {
                Link to 9,
        }
}
```

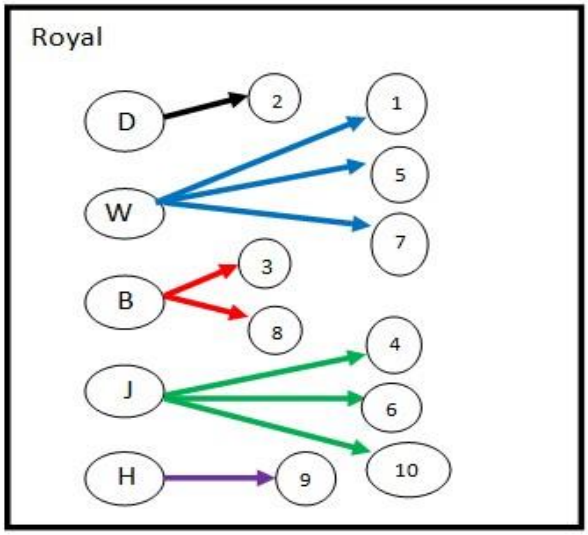


Figure 3.13 (B) Sanitized FIASSEC network syntax and Graph, called Royal, Using variable sanitization.

3.10 Deletion Method Applied to Network Syntax

All techniques up until the point that this point permitted the outsider or foe access to all of the sanitized network information. This erasure strategy for sanitization expels the information keeping it from being imparted to the substance. The quick issue with this strategy is that data can't be haphazardly erased from network data. The purpose behind this is because of the interrelationship between the nodes and the connections. In a medical database, we can expel or conceal a solitary patient's data without irritating different records. For network information, in the event that we wish to cover the presence of a connection between a couple of nodes and expel one link_ to a section for a node, we should likewise locate the relating passage for the other node and erase it too. Else, there will, in any case, exist a connection between the two nodes. Rather than

being bi-directional, however, this residual link will be directional since we have erased just a single connection _ node. Besides, on the off chance that we need to erase a node from the diagram, we should likewise recognize and erase any link_ to passages that different nodes in the chart have to the one node set apart for cancellation. Shockingly, this erasure may make the diagram be separated. This adds a level of multifaceted nature to ensure that the relating sections are additionally erased and does not secure the protection of the other data that is as yet being shared.

With this strategy we may adjust the aftereffects of the investigator of the network data along these lines, giving up expert for protection. The size or order or both will diminish in value depending on what parts are being erased. Figure 3.9, outlined the connection between diagram traits. By rolling out improvements to size and order we are influencing the length, the adjacency, the frequency and degree with conceivable effect to the various graph qualities. To limit the effect of erasure on characteristics, while erasing a node, just a node of degree 1 ought to be considered for deletion. Connections can be considered for erasure if the node occurrence is to have a degree more prominent than 1; accordingly, when the connections are erased, the graph does not end up separated.

Another minor departure from this technique is to share just the node_ name passages and erase the majority of the link_ to sections. This erasure procedure is simpler to execute since no criteria should be checked. With this strategy none of the connection between the nodes is notices. This will secure all the network attributes, except the network's order, from the enemy. Unfortunately, the expert won't have the capacity to decide anything valuable from the data except for the order of the network.

Example: The erasure of Boye from FIASSEC will bring about the deletion of the node Boye alongside every last bit of it link_ passages and well as comparing link_ to sections to Boye from

some other node in the graph. Since Boye has a degree of 1, the effect to the characteristic of the graph is insignificant. The order and size each decrease by 1. Adjacency and incidence of the remaining node is unchanged only the degree of one node; in this case, Sele decreases by one as a result of deletion of Boye. Length and distance of the remaining node are unchanged for the diameter is decreased by one due to the removal of Boye. See figure 3.14.

```
Mesh network FIASSEC{

        Node Sele{
                Link to Awoin
                Link to Sidney
        },
        Node Awoin{
                Link to Sele
                Link to Sidney
        },
        Node Sidney{
                Link to Sele
                Link to Awoin
                Link to Narobi
        },
        Node Narobi{
                Link to Sidney
        },
}
```

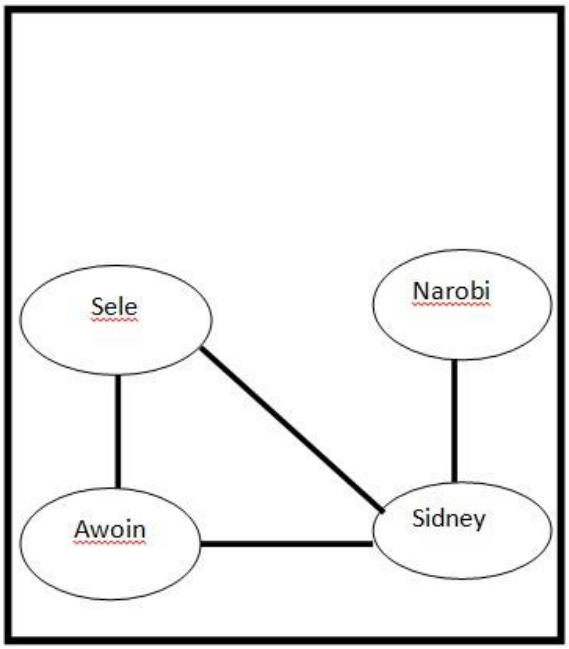


Figure 3.14 Deletion of Boye from the FIASSEC graph.

Example: Erasure can make a detached diagram in the event that it isn't performed accurately. The connection amongst Sidney and Narobi ought not to be erased since Narobi has a degree of 1 on except if we are additionally anticipating erasing Narobi node. Otherwise, the subsequent diagram will be separated since we will have wiped the main connection amongst Narobi and whatever is left of the graph. See figure 3.15.

```
Mesh network FIASSEC{
        Node Boye{
                Link to Sele
        },
        Node Sele{
                Link to Boye
                Link to Awoin
                Link to Sidney
        },
        Node Awoin{
                Link to Sele
                Link to Sidney
        },
        Node Sidney{
                Link to Sele
                Link to Awoin
        },
        Node Narobi{
        },
}
```

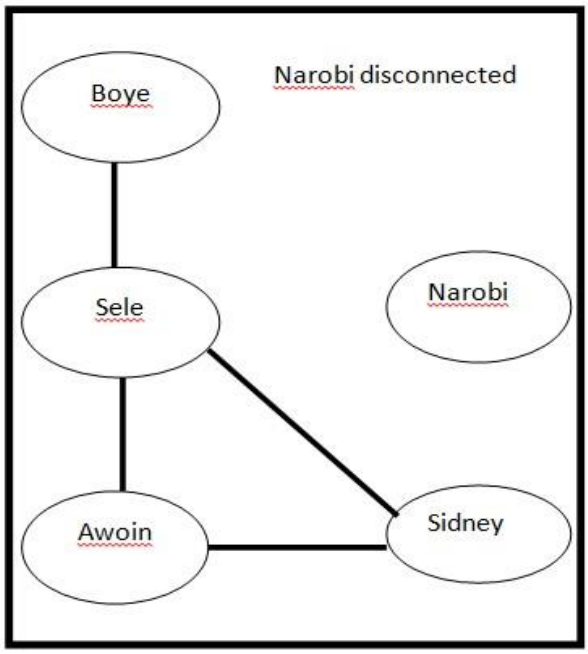


Figure 3.15 Deletion of Narobi link causing the graph to be disconnected.

Example: erasure of a link between two nodes can effectively conceal the connection amongst nodes and not disengaged the graph if the nodes both have degree > 1. Erased the connection amongst Sidney and Narobi is tricky since Narobi has a degree of one. The connection amongst Sele and Sidney, notwithstanding, can be erased since the two nodes have a degree more than 1. See figure 3.16. This deletion covers the presence of a link between these two nodes

```
Mesh network FIASSEC{
        Node Boye {
                Link to Sele
        },
        Node Sele {
                Link to Boye
                Link to Awoin
        },
        Node Awoin {
                Link to Sele
                Link to Sidney
        },
        Node Sidney {
                Link to Awoin
                Link to Narobi
        },
        Node Narobi{
                Link to Sidney
        },
}
```

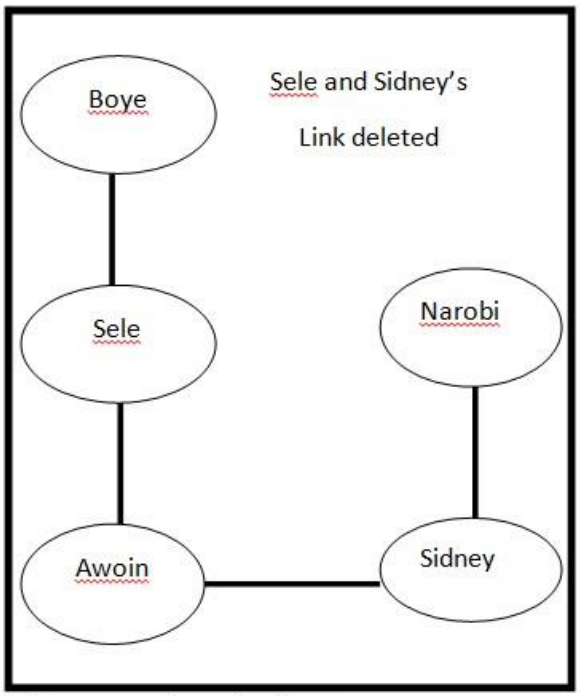


Figure 3.16 Deletion of the link between Sele and Sidney.

This current graph's size continues as before while its order is diminished by one. Since a connection expelled, the adjacency of the nodes has been influenced. The degree of Sele and Sidney has been reduced by 1. With the evacuation of this connection, the length and distance between a portion of the nodes have expanded which impacts the diameter of the diagram by extending it from 3 to 4.

3.10.1 Deletion Method Applied to Complex Network (RedIRIS)

The initial RedIRIS network as shown in figure 3.9, a network which is comprised of 18 territorial nodes and a focal node interconnected by a mesh network was subjected to the deletion method to see how good this sanitization method will work on a complex network. The syntax to achieve this is as follows as well as the resulting graph shown in figure 3.17.

Syntax mesh network RedIRIS { node Galicia { link to Asturias, link to Pais Vasco, link to Castilla Leon }, node Asturias { link to Galicia, link to Canabria }, node Cantabria { link to Asturias, link to Pais Vasco }, node Pais Vasco { link to Galicia, link to Cantabria, link to Navarra }, node Navarra { link to Pais Vasco, link to Aragon }, node Aragon { link to Navarra, link to La Rioja, link to Cataluna }, node Cataluna { link to Aragon, link to Valencia, link to Baleares }, node Baleares { link to Cataluna, link to Valencia }, node Castilla Leon { link to Galicia, link to La Rioja }, node La Rioja { link to Castilla Leon, link to Aragon }, node Valencia { link to Cataluna, link to Baleares, link to Murcia, link to Andalucia }, node Murcia { link to Valencia, link to Andalucia }, node Andalucia { link to Murcia, link to Valencia, link to Castilla La Mancha, link to Extremadura, link to Las Palmas }, node Las Palmas { link to Tenerife, link to Andalucia }, node Tenerife { link to Las Palmas, link to Madrid }, node Madrid { link to Tenerife }, node Extremadura { link to Andalucia }, node Castilla La Mancha { link to Andalucia }

}

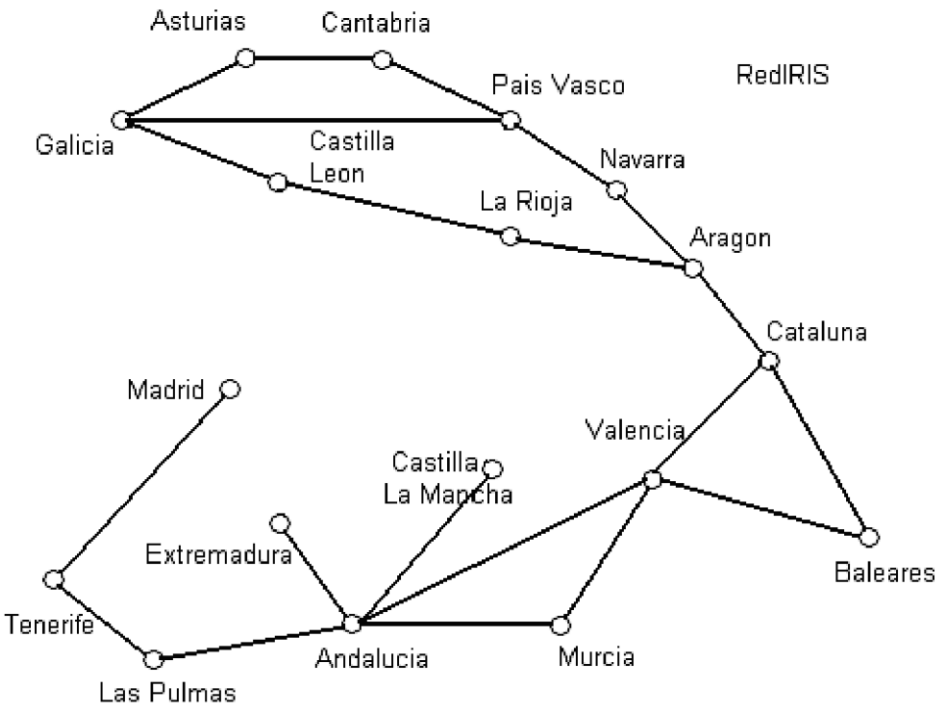


Figure 3.17 Graph after the deletion of the central node “Nodo Central”.

3.11 OTHER COUNTERMEASURE

As part of ensuring a successful aim of this study. Patches of Operating System of each of the client machines on the virtual platform was downloaded from the internet, and was installed on them to update the system to tighten the security of the operating system, as well as those client machines which needed to be updated, were upgraded for them to perform or give out their maximum performance. These processes are shown in figure 3.18 andfigure3.19. .

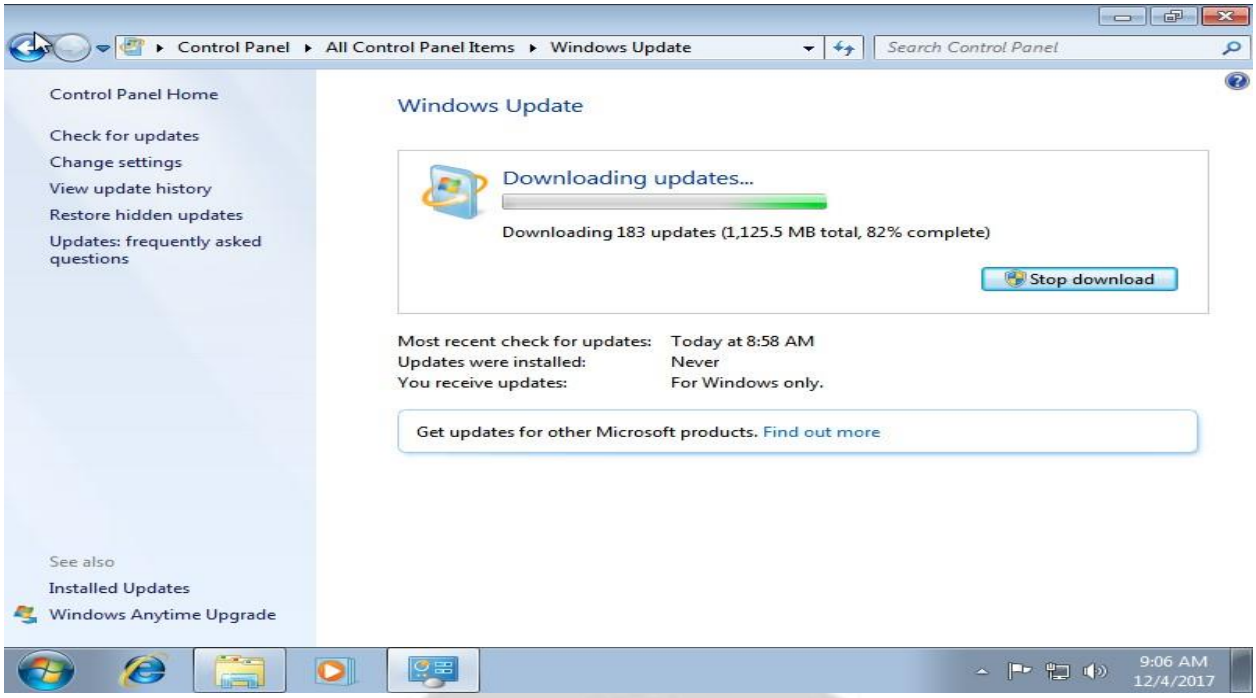


Figure 3.18 A screenshot showing patches being download from the internet.

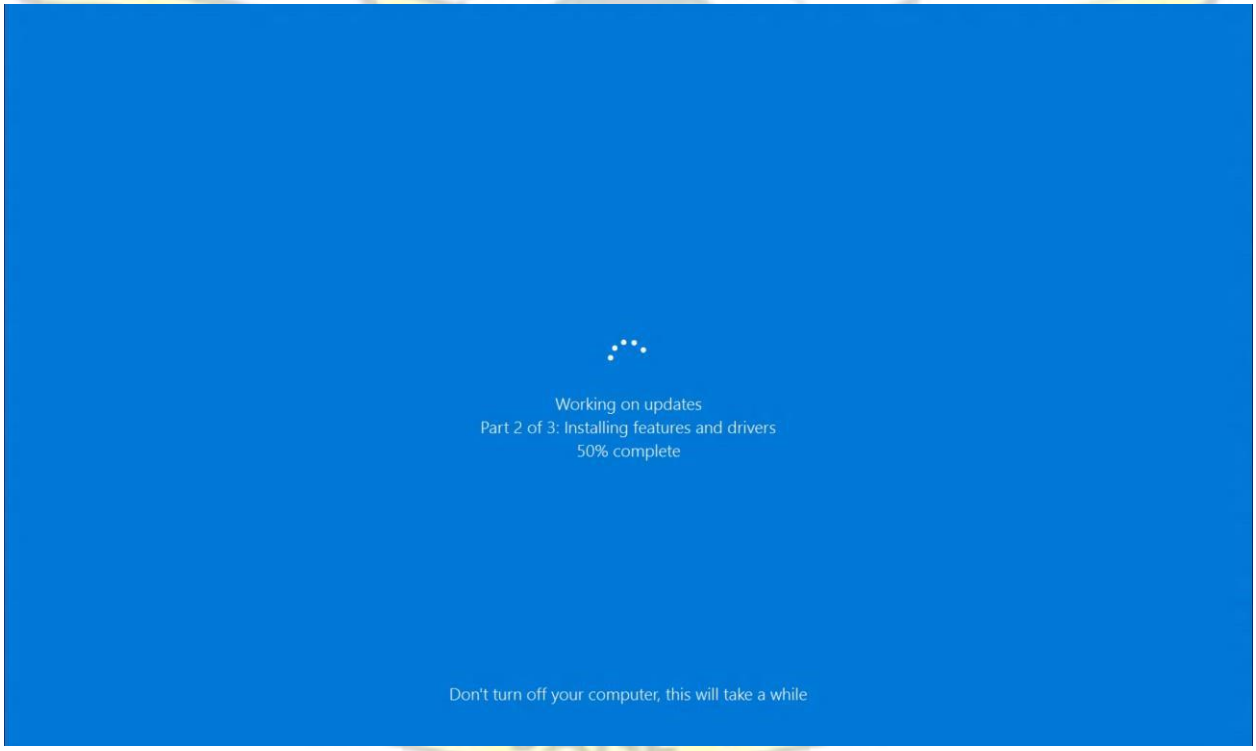


Figure 3.19 A screenshot showing patches being installed.

Also, to avoid IP spoofing attacks, all configurations on the computers that allows the computers to share status information with others on the internet was disabled to disallow all ping commands on the machines as shown in figure 3.20

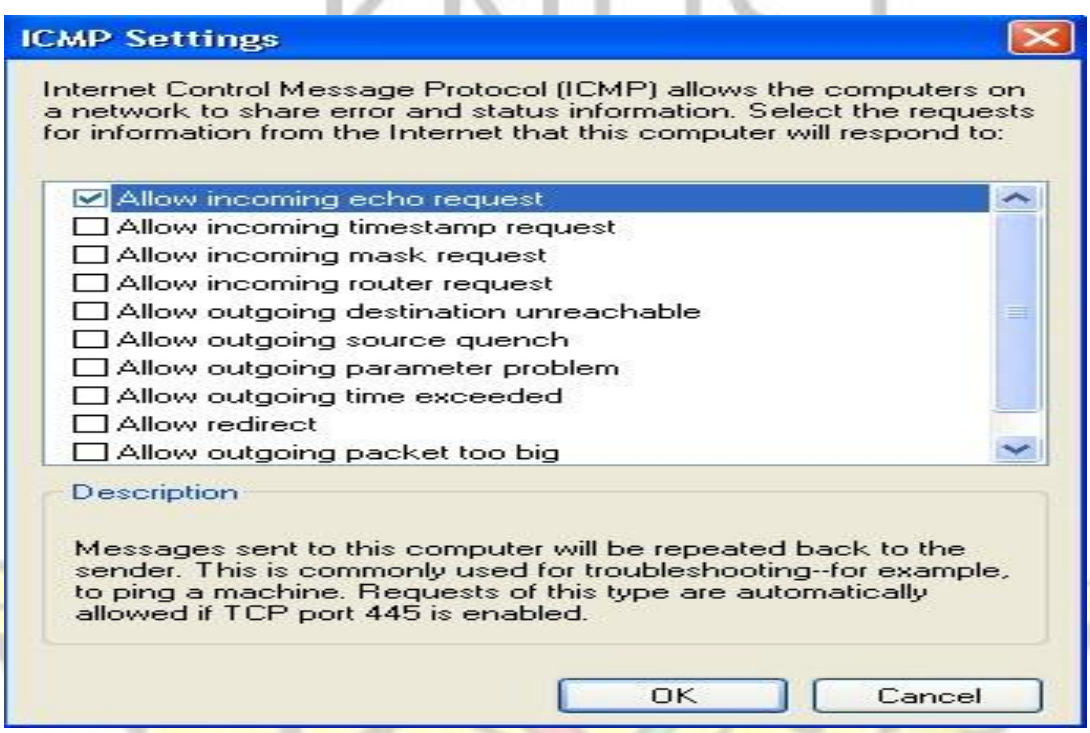


Figure 3.20 A screenshot showing disabling all ping command requests.

Finally, multilayered firewalls of the operating systems was check and configured to help protect the system on the network as shown in figure 3.21.

Figure 3.21 A screenshot showing firewalls configured

3.12 ASSURANCE TESTING

The countermeasures implemented as an antidote to the vulnerabilities of the network unearthed during the network scanning penetration test.

This test which has been codenamed “Assurance Testing” is a repetition of the previous penetration test that revealed the network vulnerabilities to check for assurance of the low or non – existence of those vulnerabilities after the implementation of the sanitization methods and the countermeasures.

Here the vulnerabilities scanning was repeated, and the result is shown in figure 3.22

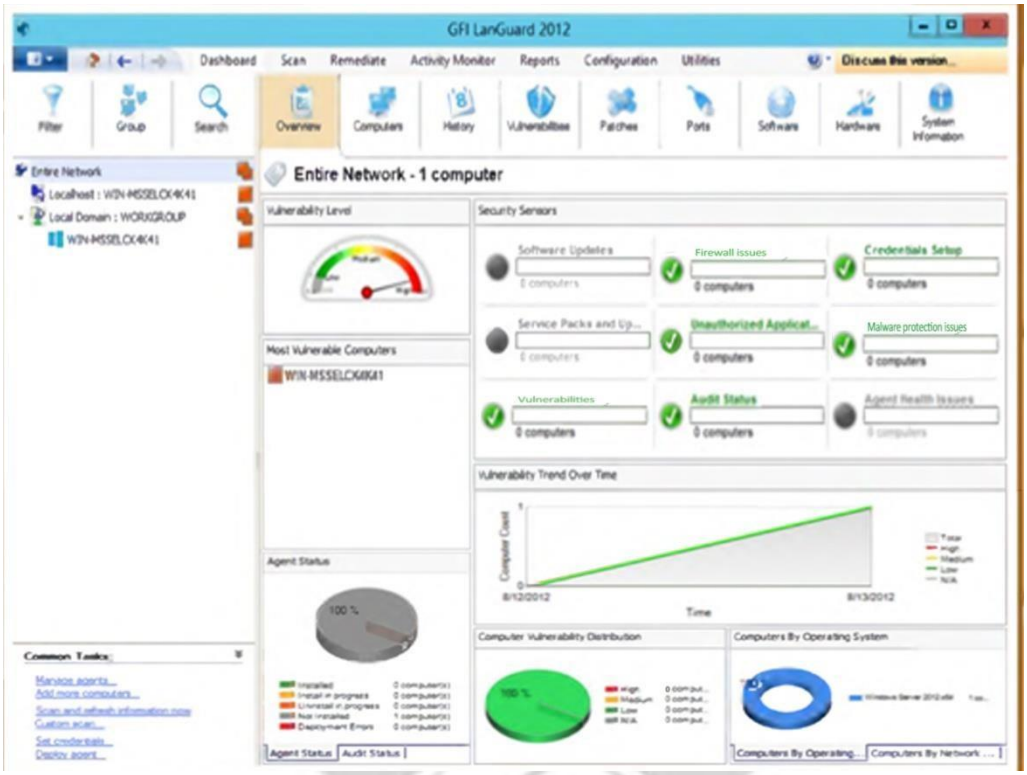


Fig 3.22 A GFI LanGuard screenshot showing Vulnerabilities scan result 2.

3.12 DETECTING REESILIENCY AFTER SANITIZATION

In view of that, a simulation was carried out on the sanitized network to detect the resiliency of the network.

Distinguishing nodes with a degree of one is a basic measurement for an expert to use to decide the effect on resiliency. Here the sanitized FIASSEC network called Y as shown in figure 3.23 was subjected to a simulation where the link between node D and W or H and J with a degree of one was cut off or disconnected from the network. After which a message was sent to the client computer whose link was detached from the network, and it was realized that the message could not get to its destination because the client computer was having only one link (degree of one) and the resulting network is shown in Figure 3.24.

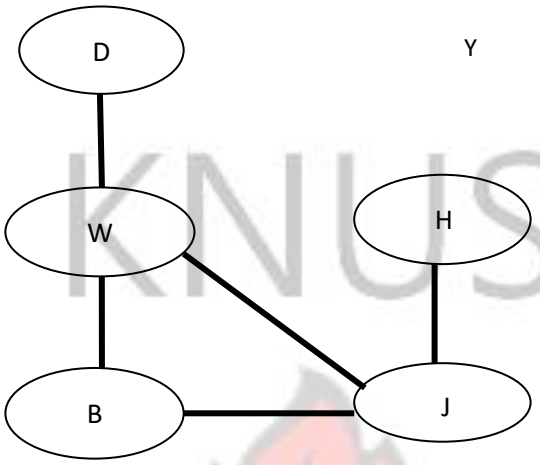


Figure 3.23 Sanitized FIASSEC network Y.

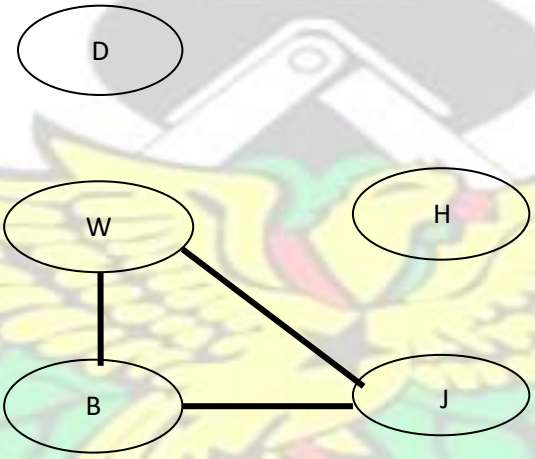


Figure 3.24Sanitized FIASSEC network Y with some of it links cut off.

A more difficult measurement that an analyst can accumulate is all part between all sets of a node to distinguish if there is a bottleneck in the network. In the simulation work, the complex network (Red IRIS) was also subjected to simulation to detect if there is any resiliency issue, even though none of the nodes are of degree 1 as shown in figure 3.25.

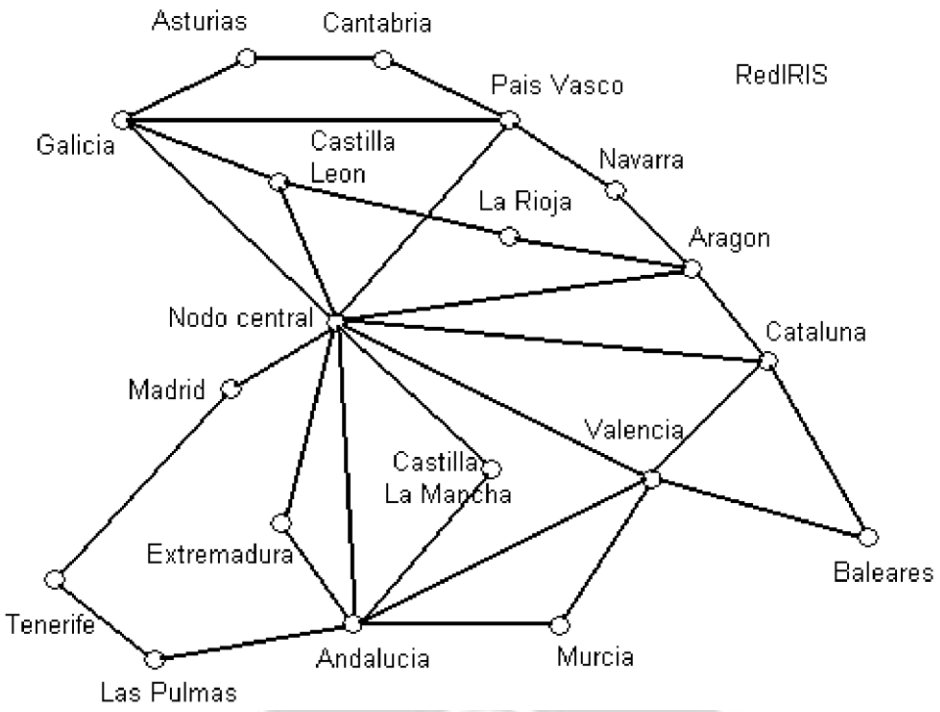


Figure 3.25.Graph of the RedIRIS network.

In the simulation, Here the node called node Central is a bottleneck or the central node from which almost all the rest of the node connected to in the network was cut off or disconnected from the network. After which a message was sent to the client computer whose link was detached from the network, and it was realized that the message could not get to its destination and the result is shown in Figure 3.26.

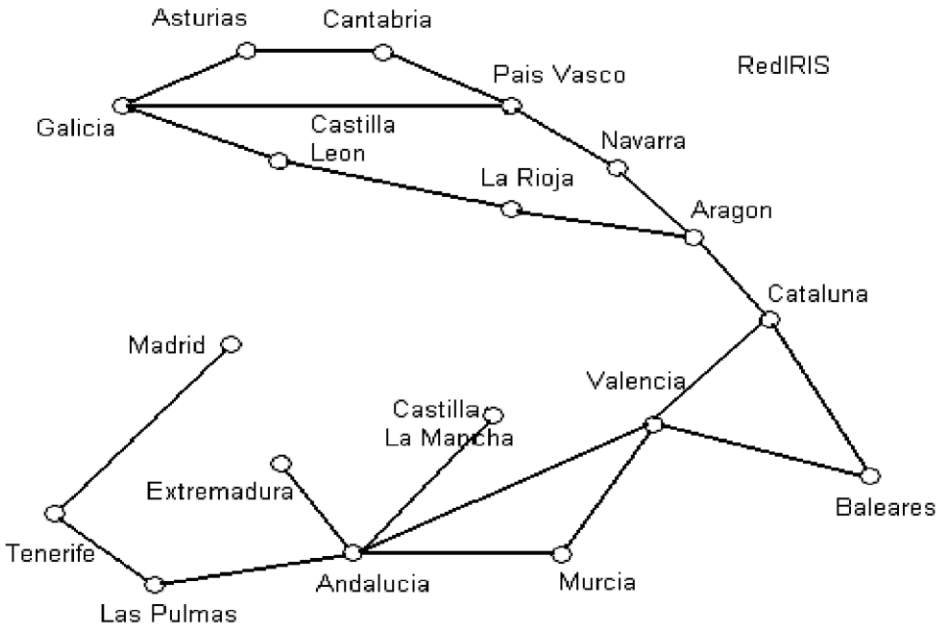


Figure 3.26. Graph after the deletion of the central node “Nodo Central”.

CHAPTER FOUR RESULTS OF THE STUDY

4.0 Introduction

In order to achieve the purpose of this thesis, many methodologies were defined and applied to the network topology in chapter three. Here in chapter four, the various methodologies defined and applied to the network topology results are being discussed as follows:

4.1 Penetration Testing Discussion

The test was aimed at checking or verifying the degree of vulnerability of the network under investigation.

As discussed in the previous chapter, a series of penetration test were undertaken to achieve the objective of the penetration test.

Notable amongst the penetration test include but not limited to the following; identification of host discovery, discovering open ports coupled with its associated services as well as grabbing system banners from a remote location.

4.1.1 HOST DISCOVERY

As illustrated in figure 4.1, Host scan was successful, and the result is shown in the picture. Hence the live hosts are the IP addresses shown on the right side of the outcome, and their details are appearing at the left side as well. Here attackers can use the result to calculate the student's marks using subnet Marks calculator to identify the number of hosts present in the subnet as well as use ping sweep to create an inventory of live system in the subnet.

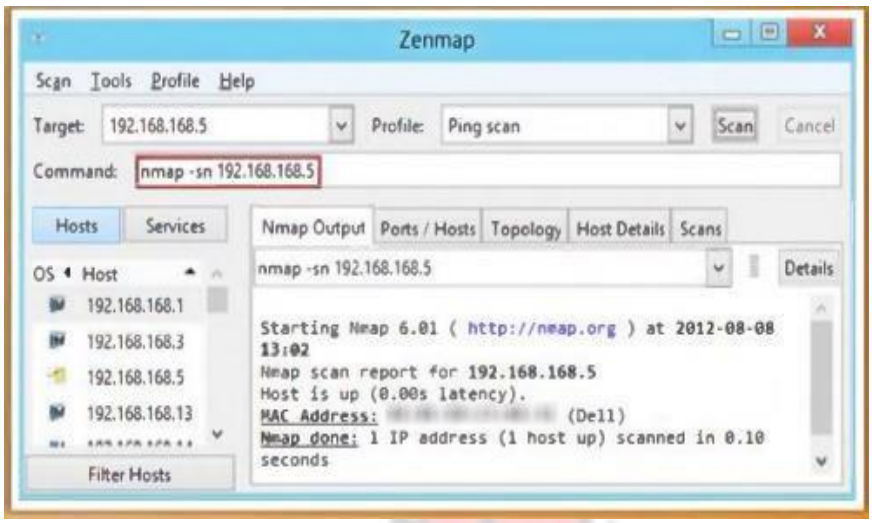


Figure 4.1 A Zenmap screenshot showing Ping scan output.

4.1.2 PORT SCANNING

In the port scanning result, it was deduced that all the ports on the network which are opened are highlighted in the green color as illustrated or seen in the result in figure 4.2.

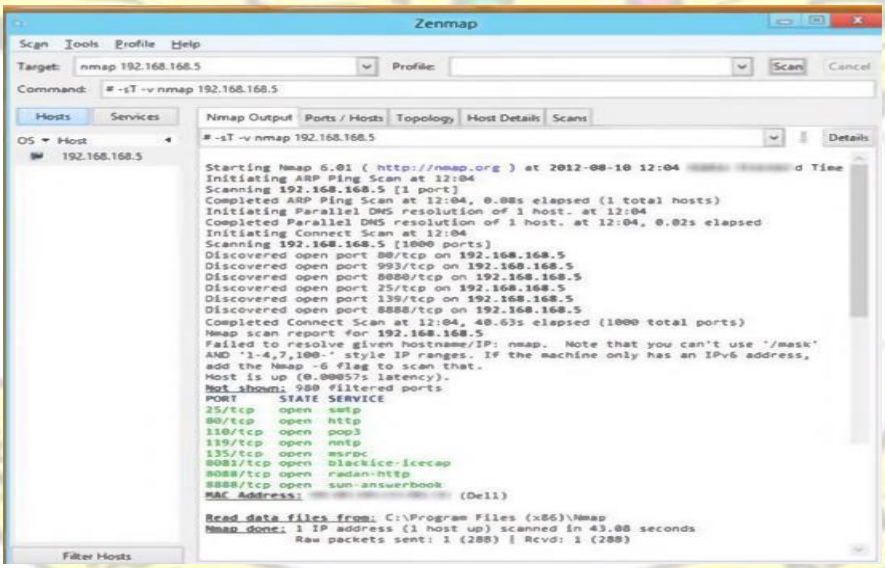


Figure 4.2 A Zenmap screenshot showing port scan results.

4.1.3 BANNER GRABBING/ O.S FINGER PRINTER

Here names of operating systems running of the remote target system were detected and shown in the result in figure 4.3 with their respective IP addresses. It helps hackers to exploit the vulnerabilities specific to that operating system.

Fig 4.3 A Netcraft screenshot showing a Banner Grabbing scan result.

4.1.4 VULNERABILITIES SCANNIG

Here color coding is used to determine the vulnerabilities in a system. As shown in the result in figure 4.4, the issues in the red color are of high risk, followed by the green of low risk and lastly the gray which is with no risk.

Fig 4.4 A GFI LanGuard screenshot showing Vulnerabilities scan result.

4.1.5 DRAW NETWORK DIAGRAM

Here it automatically discovers and creates a network map of the target network. It was able to display in-depth connections like OSI layer 2 and layer 3 topology data such as displaying switch to switch, switch to node and switch to router connection. It allows the user to perform inventory management of hardware and software assets. Shown in figure 4.5.

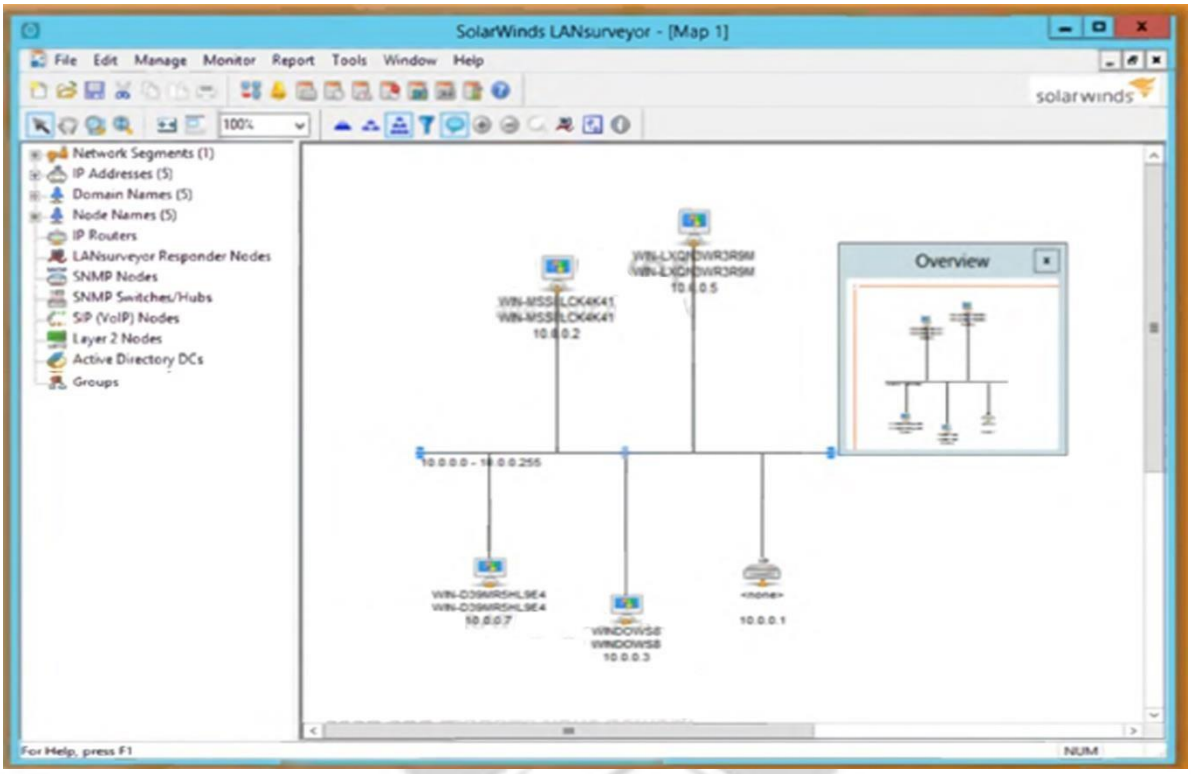


Fig 4.5 A LANsurveyor screenshot showing Network Diagram result

These modular or physical penetration tests were undertaken in a coordinated manner to ascertain the various security threats the network is bound to experience when malicious attackers employ such means of attacks.

4.2 Fixed sanitization

This method of sanitization on a network topology aims at preserving all the statistics of the network as discussed in the earlier chapters which include the connectivity, size, order, adjacency, incidence, degree, completeness, length and the distance and the uses of these estimations to recognize different qualities, for example, resiliency which is talked about in chapter three.

After applying this method to our unsanitized network (i.e., Fiassec network as well as the RedIRIS) which resulted in achieving a sanitized Fiassec network called Y and that of the RedIRIS as illustrated in figure 4.6 and figure 4.7 below.

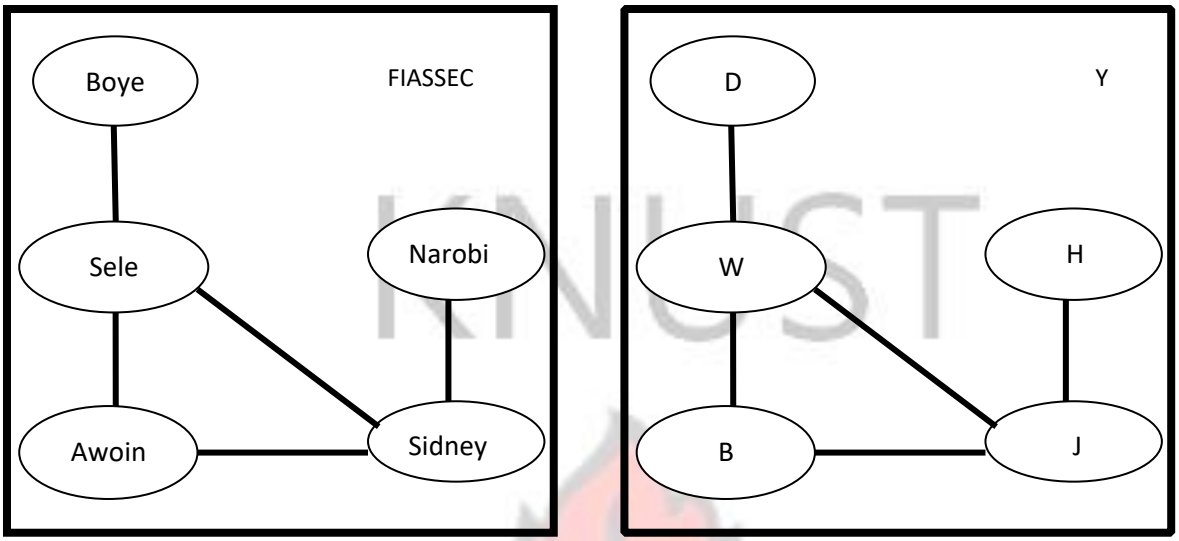


Figure 4.6. Original FIASSEC network and sanitized FIASSEC network, called Y, using fixed method.

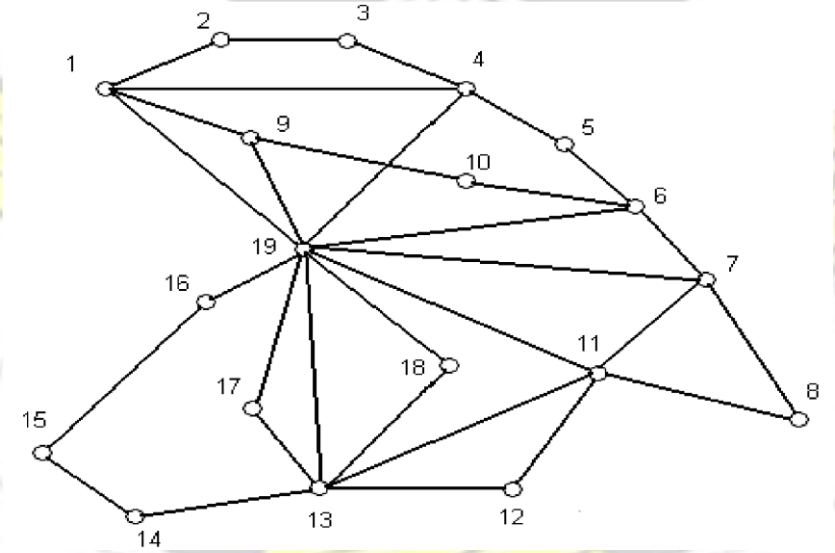


Figure 4.7 Resulting graph from fixed transformation of RedIRIS network.

It was revealed that all the statistics that were gathered on the sanitized network (resulting Fiassec network and the RedIRIS network) had yielded the same outcomes as if they were performed on the unsanitizedFiassec and RedIRIS network. In addition, this technique boosted statistical accuracy in the sanitized network at the expense of security since the central part of this network that was ensured was the names that recognize that nodes and the network. In conclusion,

for a concrete confirmation to be realized and accepted on this fixed method, this methodology was applied to a complex mesh network called the RedIRIS, and the result as illustrated in figure 4.7 was realized which was in tandem with the aims of the method.

4.3 Variable Sanitization with Separate Mappings

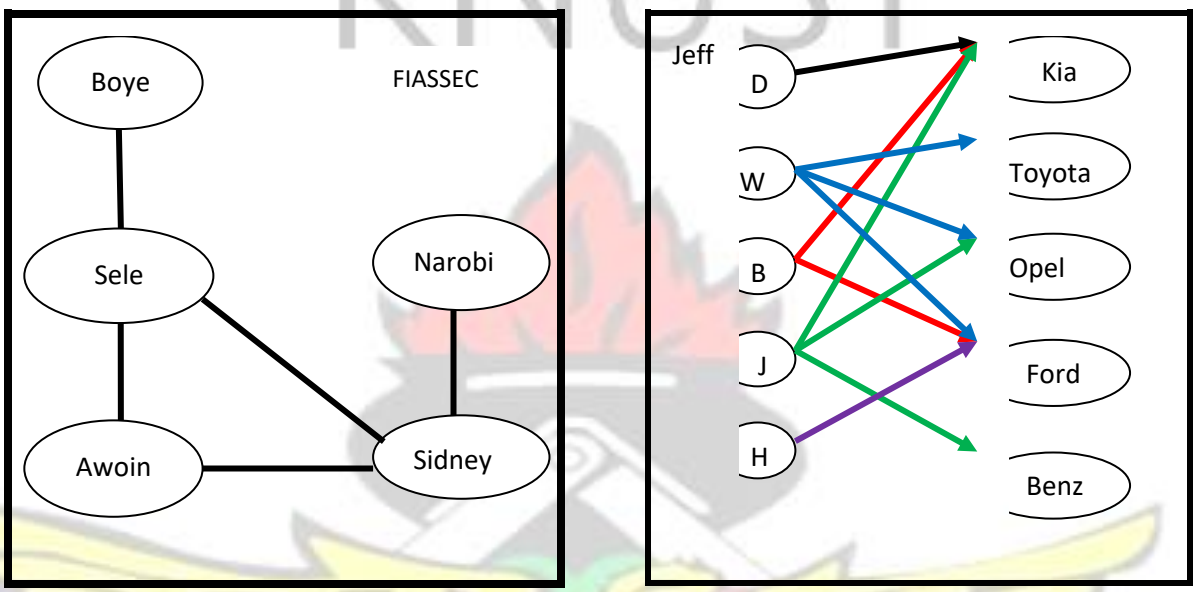


Figure 4.8 Original FIASSEC network and sanitized FIASSEC network, called Jeff, using variable sanitization.

With this type of sanitization, after applying it to the unsanitized network from now on referred to as Fiassec network where only the node name and the link to the names changes, the resulted system is the Jeff network as illustrated in figure 4.8. Furthermore, since only the nodes names were changed, it was observed that the degree of each node, the adjacency, and the incidence are the same as the original network. So here also the privacy of the node names are protected as in the fixed method.

An additional observation made was that two transformations of the Jeff network could be reconstructed by the adversary as illustrated in figure 4.9.

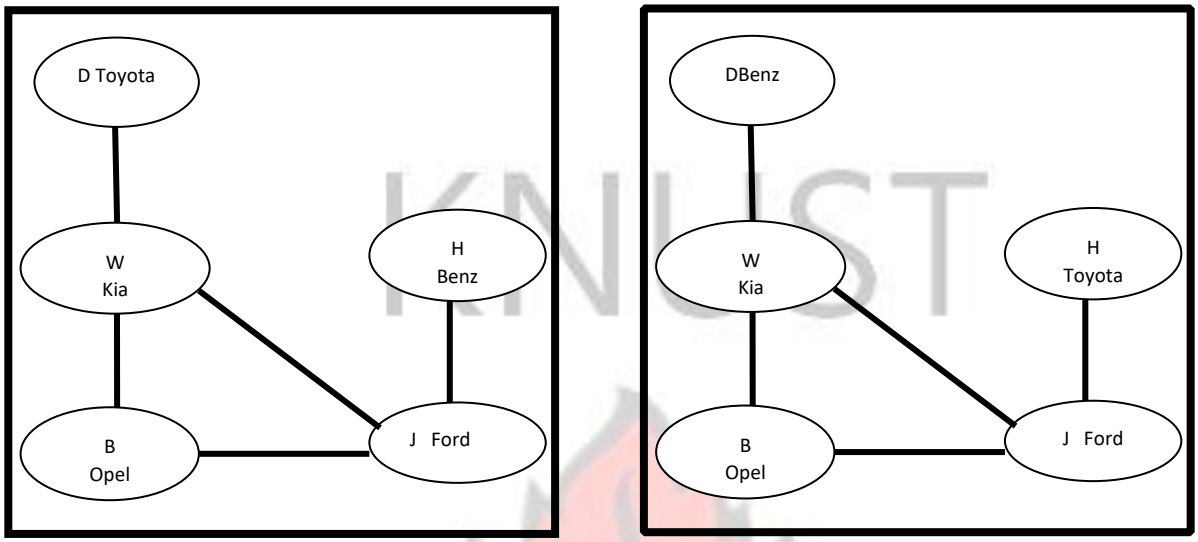


Figure 4.9. Two transformations of the Jeff network

which rather goes a long way to add a level of complexity to the variable method in the sense that the adversary finds it more difficult to deduce which of the two networks can the original Fiassec network be deduced from.

In a nutshell, it may be deduced that, with this method the existence of resiliency issue can still be determined meanwhile it also preserves measurable outcomes while expanding the level of security insurance and above all it poses a significant challenge to the adversary not been able to know the correct position of every node with relationship to alternate nodes after they have been able to reproduce the overall structure of the network as illustrated in figure 4.9.

4.4 Variable Sanitization with unique name

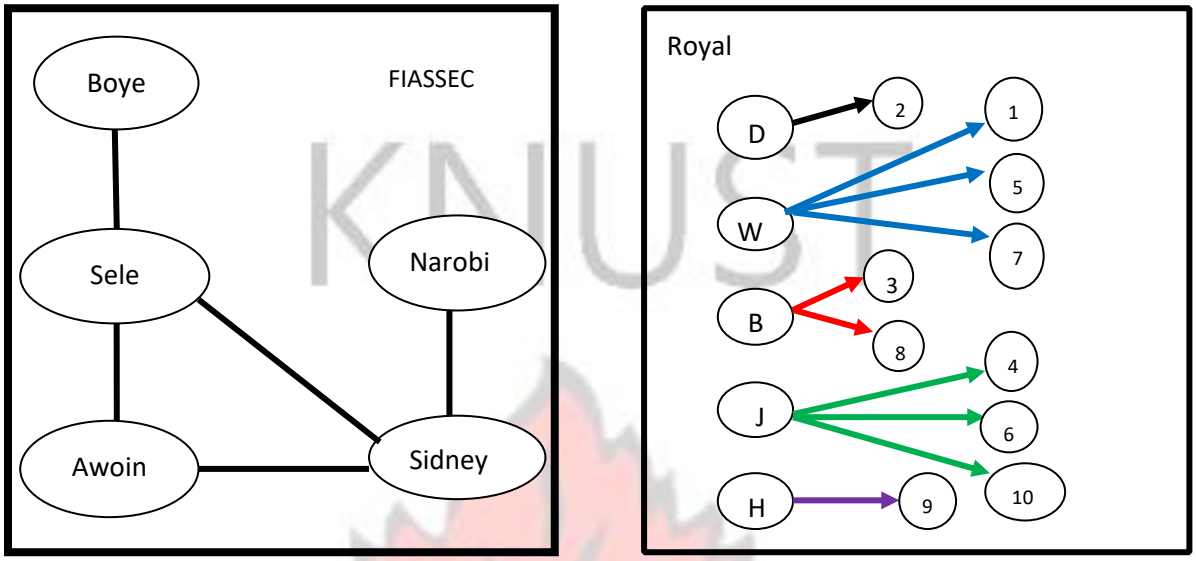


Figure 4.10 Original FIASSEC network and sanitized FIASSEC network, called Royal, Using variable sanitization.

Here in this sanitization method, the graph under study was reconstructed into a directional graph which is a disconnected with its order three times bigger and its size twice as large as the first graph as illustrated in figure 4.10 with the code name Royal.

It was also realized that this technique is more difficult than the past two strategies because of the unique mapping highlight and the challenges associated with changing the resulted Royal graph which is directional and disconnected to a bidirectional and a connected graph.

Also, it was deduced that as the quantity of nodes rises in the network, the reconstruction of the network becomes more complicated.

Finally, since remaking isn't conceivable or not endeavored because of the complex nature of the outcomes after the sanitization, the main insights that can be resolved are the size, order, and degree. The resiliency "single point of failure" can be distinguished since the degree can be resolved, but the good news is that it bottlenecks cannot be calculated.

Taking everything into account, the variable sanitization with unique name endeavors to expand the level of protection to the detriment of safeguarding measurable outcomes in the network topology.

4.5 Deletion Method Applied to Network Syntax

With the deletion method, it aims to increase protection to the expense of measurable (statistical) safeguarding. In this method it was observed that the ultimate goal is to encapsulate an aspect of the network information and in this case the more you try to hide the network information using this deletion method, the more you distort the statistical information such as the size, order, and degree of the network which needs to be protected instead. And making changes to this statistical information may go a long way to affect the length, the adjacency and the incidence which will have a greater impact on the network.

In addition, other deduction that was made included that, for this impact to be reduced, only nodes with a degree of one (1) ought to be considered for deletion and connections can be considered for deletion if the node instance is having its degree more prominent than one (1), as to avoid the graph becoming disconnected.

Finally, the method was tested and applied to the complex network called the RedIRIS, where its central node called the Node Central was made to suffer deletion leaving several of its nodes with a degree of one (1) and also demonstrating where there are congest points (traffic) or bottlenecks in the subsequent diagram as outlined in figure 4.11.

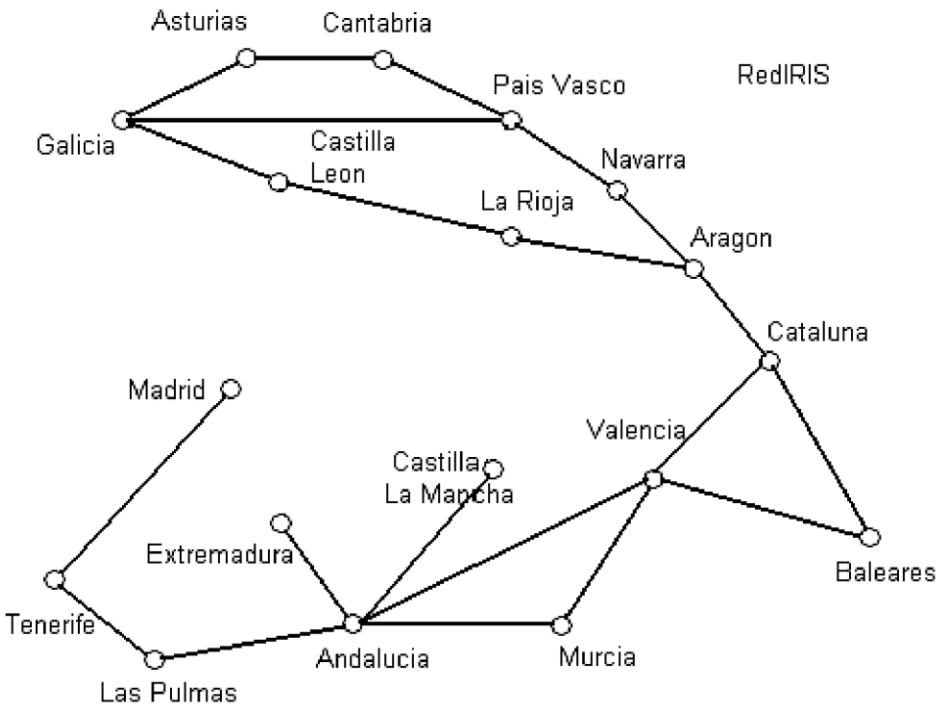


Figure 4.11. Graph after the deletion of the central node “Nodo Central”.

In conclusion, the deletion method which aims at maximizing privacy at the expense of statistical preservation has failed because it rather ends up destroying the statistical information of the network its need to protect. Although it has some advantages, it does not portray the main objections to this thesis or research work.

4.6 Detecting Resiliency After Sanitization

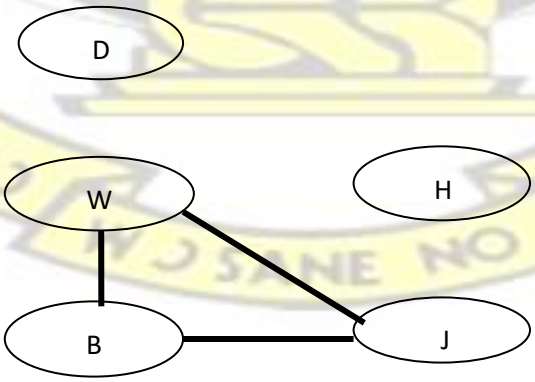


Figure 4.12 Sanitized FIASSEC network with some of it links cut off.

In the diagram above it was observed that when the link between D and W as well as the link between node h and J are broken down or disconnected, the nodes D and H are cut off of the whole network.

A remedial game-plan ought to be taken, having analyzed this resiliency, for instance, observe figure 4.12. Where the D node has a level of one, it might be conceivable to include a connection between D what's more, it to build the degree of D and H over one. If the expansion interface isn't conceivable, then it might be plausible to move a connection starting with one set of a node then onto the next. As such, move the connection W and J to D and H which still make a network with a similar number of nodes and connections, however no node of a degree of one to bring about system versatility. Shown in figure 4.13

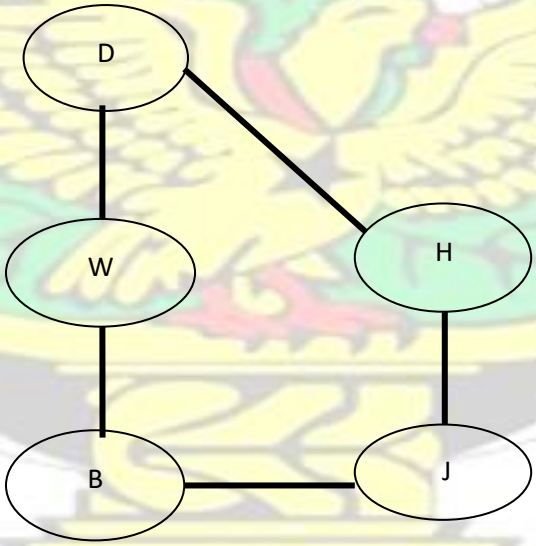


Figure 4.13 Resolving resilience issue by moving a link from W and J to D and H. With the network pictured in figure 4.14,

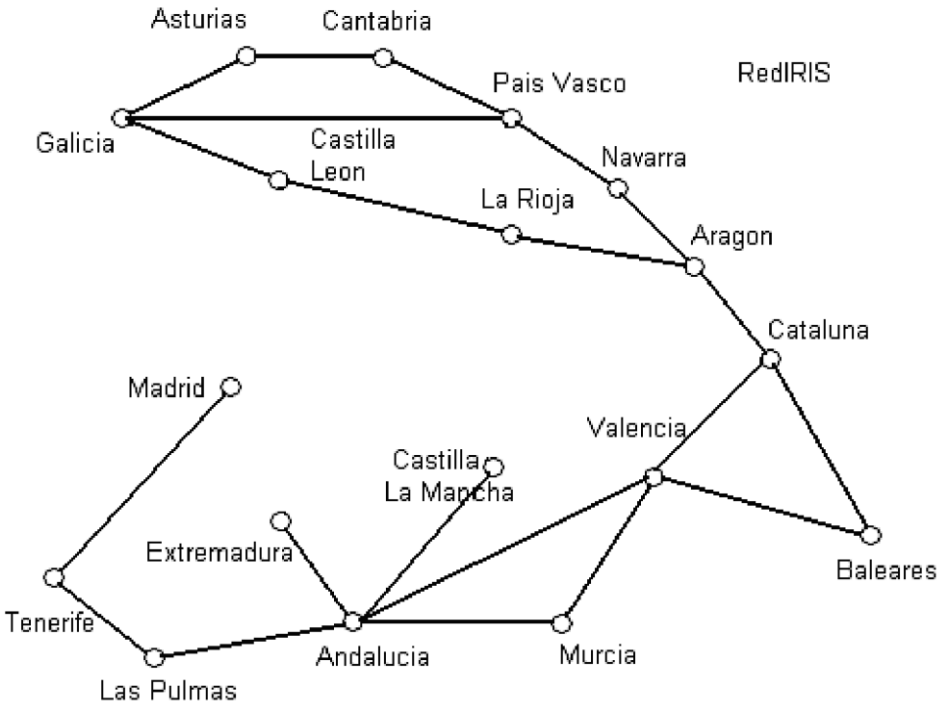


Figure 4.14 Graph after the deletion of the central node “Nodo Central”.

One simple approach to determine the resilience issue is to make a connection between the nodes with degree one. That is Madrid, Extremadura and Castilla La Mancha which at any time can be disconnected from the network when the link between them and the other nodes are broken illustrated in figure 4.15, where two additional links are being added to resolve the resilience issue in the network.

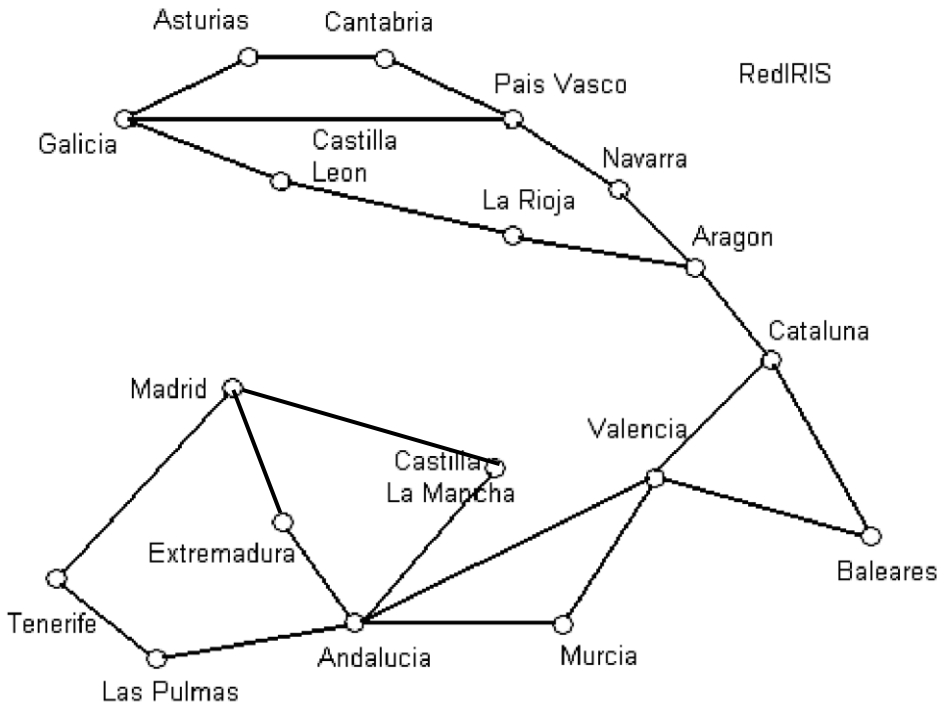


Figure 4.15 Resolving resilience issue by adding links between Madrid and Extremadura and between Madrid and Castilla la Mancha.

4.7 Assurance Testing

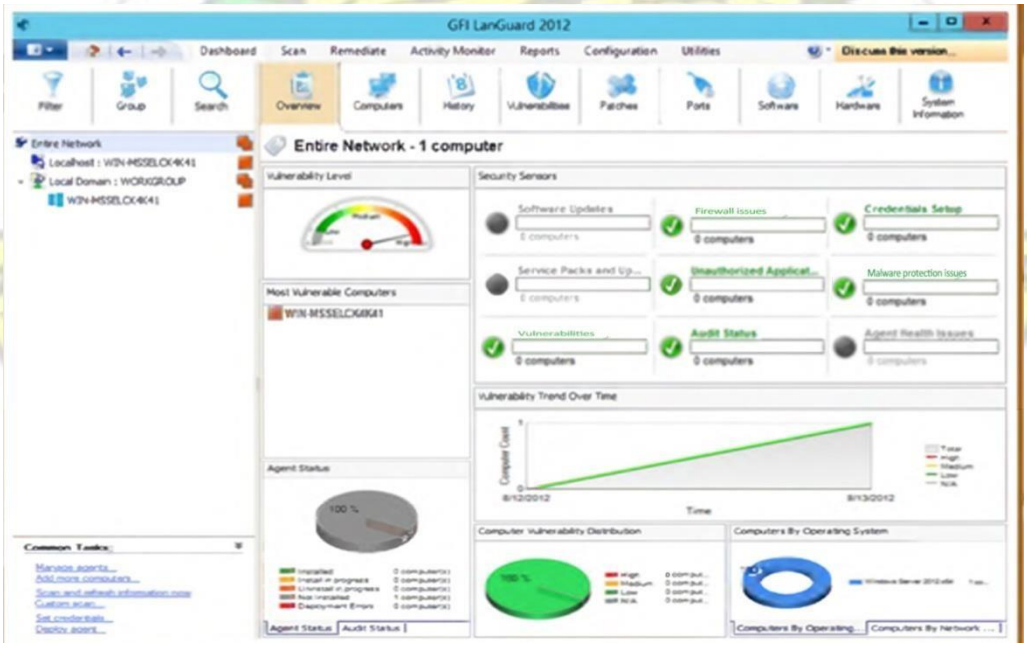


Fig4.16 A GFI LanGuard screenshot showing Vulnerabilities scan result 2.

Here it was deduced that the system has very low vulnerabilities or even non-existence of weaknesses, because from the colour coding shown in figure 4: which is the green colour representing low risk and the Gray representing no risk.

As comparing this to the figure 4.17 that is the vulnerabilities scan result before the implementation of the sanitization and the countermeasures where the system was full of vulnerabilities indicated by the red colour representing a high risk.

Fig 4.17 A GFI LanGuard screenshot showing Vulnerabilities scan result.

4.7 Summary of Discussions

Sanitizing a network ensuring, its resiliency and preserving the privacy of the network information are the foci of this research experimentation. The experiments applied methodologies, as well as the discussions done in the preceding chapters and sections, illustrate some closely related results

and as well as some practical means of ensuring or advising the stated objectives of the research project.

It is however obvious that despite the array of methodologies or techniques employed in this experiment a hybrid sanitization technique will result in the realization of the objectives of the research. This hybrid sanitization model comprises of the fixed and the variable sanitization model.

The application of the hybrid model is such that the fixed sanitization technique is first applied to the unsanitized network to preserve statistical information associated with the network such as the order, size, connectivity, adjacency, incidence, completeness, length, degree and distance. Following the application of the fixed sanitization technique is the application of any of the variable sanitization methods to ensure the resilience of the sanitized network. And as well as increasing the level of privacy protection by posing some high level of challenge to the adversary. Who would not easily know or guess the correct position of every node even after been able to reproduce the overall structure of the network.

CHAPTER FIVE CONCLUSION, RECOMMENDATION AND SUGGESTIONS

5.0 Introduction

The purpose of this study was to sanitized data to improve encapsulating particular network topology. This chapter covers conclusions, recommendation, and suggestions for further studies.

5.1 Conclusion

The essential outcome of this theory is that purifying (sanitization) influence the sensitive understanding between data security and data examination. The more moderate the purging procedures, the more precise the examination is on the post-cleansing data to the weakness of the assurance of the data. Then again, the more the powerful the sanitization procedures to grow security protection, the less exact the specific bits of knowledge are when performed on the cleaned information. Sanitization techniques are appealing and practical response to shield an enemy or outside social affair from isolating information once they get to the data. As the range of the system topology extends so do entangled natures in replicating the unrefined system data from the sanitized data. Irrelevant system and complete work frameworks are difficult to shroud using sanitizations methodologies on account of the novel characters of the topology. The language structure contemplates a standard in which any framework topology can be described. This examination can be connected with consolidate specific framework topology positions.

Finally, the ideal of this theory can be applied to all the network topologies.

5.2 Recommendation

This exploration can be stretched out to alternate sorts of neat work shown in figure 1.2.

Likewise, we can consolidate the extra node _name and connect _to property into the disinfection that we did exclude in this work since the data was unrestricted. Since credits can be particular to an association, having the capacity to join in these discretionary characteristics at an abnormal state will enable adaptability for every association to indicate the same number of the traits as required. If this data is incorporated into the sentence structure, however, sanitization techniques should have the potential to deal with this extra data without bargaining the security and investigation strategies. Counting the characteristics may make the ID of the contiguous nodes, for instance, simpler by having the capacity to combine up the connection _to properties from the hubs in the sentence structure or syntax. The connections in the syntax are indirectly bidirectional. Associations inside a network, in any case, may not all be bidirectional. Inward firewalls in a system, for instance, may require the graph to have coordinated connections. Disinfection of blended systems, that is, systems with both directional and bidirectional connections can be cleaned utilizing the techniques talked about in this research. The system language structure (syntax) talked about in section 3.3 is adaptable to be apply to coordinated, bidirectional, and blended systems. Despite the fact that lone bidirectional systems particularly examined in this research, the cleansing procedures introduced here are likewise pertinent to directional and mixed networks and can be investigated as an extension of this research.

This work can connect to a network: organization intranet, the web, a neural system, phone arrange, guide of roadways, or a chart of human associations. Any build or idea that can render as a bi-directional diagram can subject to these sanitization techniques.

5.3 Suggestions for Further Studies

Strategies, for example, the expansion of "phony" or "sham" hubs and connections can be investigated when utilized as a part of the connection with the cleansing techniques depicted in this

paper. These systems will make recreating the crude information more troublesome since the foe won't know which of the sterilized data is the actual system data and which data is phony. The expansion of sham hubs or nodes to the system will build the network size and in this way the complicated nature of the system making it harder to construct. The exchange off between arranging protection and examination with the expansion of node and connections can investigate. These strategies present false information which can unfavorably influence any investigation that performed on the report since we are currently managing data that isn't in actuality part of the real system setup. With the fixed and variable sanitization strategies we investigated, we expected the most pessimistic scenario where the foe approaches the more significant part of the post-cleansing information. Another choice to study is confining the entrance to share just a subset of the data or have an interface to limit get to straightforwardly to the majority of the information. This can be expert through utilizing sees onto a database where the network data is kept.

REFERENCES

Adjacent Vertices", from Mathworld – A Wolfram Web Resource. http://mathworld.wolfram.com/AdjacentVertices - Weisstein

Bernard Kolman, Robert Busby, Sharon Ross(1996*) "Discrete Mathematical Structures"*, Third edition, Upper Saddle River, New Jersey, Prentice-Hall.

Bishop, Bhumiratana, et al. (2004) "How to Sanitize Data"

Chartrand, Zhang (2004) "Introduction to Graph Theory".

Cohen, F. (1998) A Note on the Role of Deception in Information Protection, Computers & Security, 17: 483-506.

Computer Networking | Wi Fi | Node (Networking) https://www.scribd.com/presentation/242294269/Computer-Networking

Dewar, M. (1989) The Art of Deception in Warfare, David & Charles, London.

D. K. Kamran Ahsan (2002). "Practical Data Hiding in TCP/IP. Proc". Workshop on Multimedia Security at ACM Multimedia,

Dinur, Nissim (2003) "Revealing information while preserving privacy".

Eric Weisstein, *"Adjacent Vertices"*, from Mathworld – A Wolfram Web Resource. http://mathworld.wolfram.com/AdjacentVertices

Eric Weisstein, *"Connected Graph"*, from Mathworld – A Wolfram Web Resource. http://mathworld.wolfram.com/ConnectedGraph.html

Eric Weisstein, *"Graph"*, from Mathworld – A Wolfram Web Resource. http://mathworld.wolfram.com/Graph.html

Eric Weisstein, *"Graph Distance"*, from Mathworld – A Wolfram Web Resource. http://mathworld.wolfram.com/GraphDistance.html

Eric Weisstein, *"Graph Edge"*, from Mathworld – A Wolfram Web Resource. http://mathworld.wolfram.com/GraphEdge.html

http://www.DataMasker.com/datascramblingissues.pdf

Fowler, C., Nesbit, R. (1995) Tactical Deception in Air-Land Warfare, Journal of Electronic Defense, 18(6):37-44.

Fred Horney, "*About BNF Notation*", http://bluehawk.monmouth.edu/~fhorney/SE306/About%20BNF%20notation.htm

Gary Chartrand (1985) *"Introduction to Graph Theory"*, Mineola, New York, Dover Publications.

Graph Diameter", from Mathworld – A Wolfram Web Resource. http://mathworld.wolfram.com/GraphDiameter.html - Weisstein

Graph Distance", from Mathworld – A Wolfram Web Resource. http://mathworld.wolfram.com/GraphDistance.html - Weisstein

GLOSSARY UNITS 1&2 Flashcards | Quizlet https://quizlet.com/107348503/glossary-units-12-flash-cards/

Graph Edge", from Mathworld – A Wolfram Web Resource. http://mathworld.wolfram.com/GraphEdge.html - Weisstein

Heinz Knutzen, "*Description Language for Security Policy and Topology*", http://netspoc.berlios.de/language.html

IritDinur, KobbiNissim (2003) *"Revealing Information while Preserving Privacy"*, Proceedings of the22nd ACM SIGMOD-SIGACT symposium on Principles of database systems, San Diego, California,Unites States, Pages: 202 – 210. http://portal.acm.org/citation.cfm?id=773173#

R. J. Anderson and A. P. Petitcolas, (May 1998) "On the limits of steganography" IEEE Journal on Selected Areas in Communications, vol. 16, no. 4, pp. 474--481.

Rick Eary, "*Contemporary Network Architectures – Network Topologies*", http://www.vermontel.com/~rickeary/_cnasite/CNA_L06_Network_Topologies.htm

RupaliGawade, PriyankaShetye, VaibhaviBhosale, P N. Sawantdesai (February 2014) IJARCCE

(International Journal of Advanced Research in Computer and Communication Engineering Vol. 3, Issue 2,).

Rowe, N. and Rothstein, H. (2004) Two Taxonomies of Deception for Attacks on Information Systems, Journal of Information Warfare, 3(2):28 – 40.

S.J. Murdoch, S. Lewis (June 2005)"7thlriformation Hiding Workshop".University of Cambridge, United Kingdom

Steven 1. Murdoc and Stephen Lewis, (2005) "Embedding Covert Channelsinto TCP/IP". Information Hiding Workshop 2005 proceedings on,

The On-Line Encyclopedia of Integer Sequences. Published electronically at http://www.research.att.com/~njas/sequences - Sloane

U.S. Army(1988) FM 90-2 Battlefield Deception, U.S. Army, Washington.

U S. C Information Sciences Institute, (September 1981) "Internet protocol, darpa internet program, protocol specification,"Specification prepared for Defense Advanced Research Projects Agency.

Weisstein, Lambert "W-Function", From MathWorld - A Wolfram Web Resource. http://mathworld.wolfram.com/LambertW-Function.html

Winfried Grassmann, Jean-Paul Tremblay, (1996)*"Logic and Discrete Mathematics: A Computer Science Perspective"*, Upper Saddle River, New Jersey, Prentice-Hall.

ZHANG lie etc (January 2005). "Information hiding in TCPIIP based on chaos".Journal on Communication.voJ.26NO.