Project Submission Pro-Forma

(***to be bound in front of the submitted Dissertation)***

NAME: Ayner Antonio Pérez Tito

Student ID: 1793561

I wish the dissertation to be considered for (tick **one** only)

MSc in Cyber Security Management	❑
MSc in Cyber Security Engineering	✓
MSc in e-Business Management	❑
MSc in Engineering Business Management	❑
MSc in Healthcare Operational Management	❑
MSc in Innovation & Entrepreneurship	❑
MSc in International Technology Management	❑
MSc in International Trade, Strategy & Operations	❑
MSc in Management for Business Excellence	❑
MSc in Manufacturing Systems Engineering & Management	❑
MSc in Programme & Project Management	❑
MSc in Service Management & Design	❑
MSc in Supply Chain & Logistics Management	❑
MSc in Sustainable Automotive Engineering	❑

I have checked that my modules meet the requirements of the above award ✓
I confirm that I have included in my dissertation:

An abstract of the work completed	✓
A declaration of my contribution to the work and its suitability for the degree	✓
A table of contents	✓
A list of figures & tables (if applicable)	✓
A glossary of terms (where appropriate)	❑
A clear statement of my project objectives	✓
A full reference list	✓
Confirmation of ethical approval (confirmation email)	✓

The Ethical Approval reference number is …**REGO--2017-WMG-0877**

I consent to ongoing storage of this dissertation and potential access by third parties (e.g. for staff/ student training purposes) ✓

Signed: …… …… Date: 20/08/2018

Calculating Network Security Metrics

by

Ayner Antonio Pérez Tito, MSc

(1793561)

Dissertation submitted in partial fulfilment for the Degree of Master of Science in Cyber Security Engineering

WMG

University of Warwick

Submitted August, 2018

Abstract

As companies and organizations expand and grow, the communication network that interconnects their assets is prone to change. To protect the communication network and the elements inside it, security appliances and protocols are deployed to protect and enforce the security politics inside the company.

In order to acquire a suitable security solution for the needs of the company, an analysis of the network is needed. The analysis can be made by using different theories and models that asses the threats inside an organization. However, as new technologies are developed, new considerations and analysis are needed.

In the following pages, a security metric-based model is developed which allows comparing security upgrades proposed by the network or security managers considering the security appliances and threats inside the company. The development is made by considering the metrics provided by the network and other models that assess the security or vulnerability of the network.

The research is done over LAN networks which uses the TCP/IP model to allow communication between the assets of the company. The network is based on several case studies where other models and frameworks were tested.

At the end of the research, the proposed model is tested in a virtual network together with other models to compare the results and produce a critical analysis of the highlights and limitations of the model.

The results show that the model was able to consider both vulnerabilities on the communication protocols and the security already installed in the network, However, a possible future work would be considering software vulnerabilities as well.

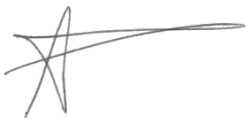
Acknowledgements

The project has been the result of a year of work and dedication, where the author must recognize the support of academics, family and classmates in which he would like to mention:

- **Harjinder Lallie:** For his help and support as a supervisor and the continuous feedback through the project.
- **Peter Norris:** For his encouragement during the experiment and his support.
- **My family:** For helping the author in his study journey
- **Andrea Sotelo:** For her feedbacks and constant support in the most difficult parts of the project.
- **Simge Koçtaş and Yao Deng:** For their feedback on my results and their constant support throughout the last part of the research.

DECLARATION

I have read and understood the rules on cheating, plagiarism and appropriate referencing as outlined in my handbook and I declare that the work contained in this assignment is my own, unless otherwise acknowledged.
No substantial part of the work submitted here has also been submitted by me in other assessments for this or previous degree courses, and I acknowledge that if this has been done an appropriate reduction in the mark I might otherwise have received will be made.

Signed candidate 

You are required to justify your submitted thesis against the degree definition for which you are registered. This needs to be downloaded and pasted into the box below.

Project definition for my degree

The project for MSc Cyber Security Engineering must:

1. address a research question directly relating to cyber security, AND
2. demonstrate understanding of the particular issues around conducting research in the cyber domain, AND
3. conduct research in the cyber domain in an appropriate manner.

My project relates to this definition in the following way:

The focus of this project is to develop a security metric-based model for measuring different security solution alternatives to help in the decision-making process of choosing a security solution and increase the overall security of a TCP/IP network. Currently, there are several studies proposing different views of how to evaluate the security, but each of them has different goals and only some of them apply to analyse possible upgrades to a network.

The project involves the understanding of different security concepts related to networks and the communication between terminals and servers. The final solution contributes to providing a tool for understanding the security needs of a network and aiding in the decision of choosing a security upgrade. Furthermore, a virtual network has been designed to test the proposed model and it can be of use for other researchers into simulating network environments.

Table of Contents

1 Background 1

1.1 Introduction 1

1.2 Research question 2

1.3 Dissertation structure 3

1.4 Motivation 4

2 Research Methods 6

2.1 Research aim 6

2.1.1 Analysis of metrics and models 7

2.1.2 Model development 7

2.1.3 Testing of the model 8

2.1.4 Analysis of the results 9

2.2 Research philosophy 10

2.3 Research approach 10

2.4 Research Design and Strategies 10

2.4.1 Literature sources 11

2.4.2 Literature review method 11

2.5 Conclusion 11

3 Literature review 13

3.1 Security metrics 13

3.1.1 Categories of metrics 14

3.2 Models to assess the level of security 16

3.2.1 The Cyber Prep methodology model 16

3.2.2 Attack criticality assessment framework 18

3.2.3 CAESARS Framework 18

3.2.4 SAEM analysis model 19

4 Model proposal 20

4.1 Model comparison 20

4.2 Establishing the inputs and outputs needed 22
4.2.1 Analysis of the inputs 23
4.2.2 Analysis of the outputs 24
4.3 Establishing the model to be proposed 25
4.3.1 Establishing the monitoring point 25
4.3.2 Establishing the Vulnerability Score 25
4.3.3 Establishing the Security Index 27
4.3.4 Analysis of security upgrades 29
4.3.5 Analysis of the procedure to understand the scores 31
5 Experimental plan 32
5.1 Generating the network environment 32
5.1.1 Tools used in the experiment 37
5.1.2 Generation of the experimental environment 38
5.2 Definition of security solutions for the network environment 43
5.3 Application of the model 45
5.3.1 Calculation of the Vulnerability Score 47
5.3.2 Calculation of the Security Index 49
5.3.3 First upgrade: Implementation of HTTPS 50
5.3.4 Second upgrade: Filtering ICMP 51
5.3.5 Third upgrade: Intrusion Prevention System 51
5.3.6 Fourth upgrade: Application Firewall 52
6 Analysis and Discussion 54
6.1 Analysis of the results 54
6.2 Comparison with other models 55
6.2.1 Results of the CAESARS Framework model 55
6.2.2 Results of the SAEM analysis method 57
6.2.3 Comparison of the results 60
7 Conclusions and Future Work 63
8 References 64
Appendix A: Consent Form 67
Appendix B: Participant Information Leaflet (PIL) 68

Appendix C: Ethical Approval confirmation 69
Appendix D: Metrics defined by Pendleton, et al. 70
Appendix E: Metrics defined by Tadda & Salerno and Duggan, et al. 73
Appendix F: SAEM analysis model 75
Appendix G: Options for network emulators 78
Appendix H: Configuration files for the experiment 80

List of Figures

Figure 1.1: Dissertation's structure 4

Figure 3.1: Adversary Levels Table 17

Figure 4.1: Diagram of inputs and outputs for the model 23

Figure 4.2: Formula to calculate the percentage of vulnerabilities in each range 26

Figure 4.3: Security Index formula 29

Figure 4.4: Security Index formula for an upgrade 31

Figure 5.1: TV company's network used in a case study 33

Figure 5.2: Example of a multistep attack 34

Figure 5.3: Topology based on a real network 35

Figure 5.4: Network topology of the experiment 37

Figure 5.5: Configuration commands to create the initial files for the experiment 40

Figure 5.6: Example of the Directory File structure of the experiment 40

Figure 5.7: Results of validating the DNS configuration 42

Figure 5.8: Validation of the web service 43

Figure 5.9: Security upgrade proposal – Intrusion Prevention System 44

Figure 5.10: Security upgrade proposal – Application Firewall 44

Figure 5.11: Monitoring point 45

Figure 5.12: Extract of packets 46

Figure 5.13: Extract of the National Vulnerability Database search webpage 47

Figure 6.1: Result of the vulnerability analysis in Web server 56

Figure 6.2: Enabling HTTPS instead of HTTP in the Web server 56

Figure 6.3: Vulnerability scanning of Web server after upgrading 57

Figure 6.4: Security coverage of the upgrades 60

List of Tables

Table 4.1: Comparison of metrics used by different models 21

Table 4.2: Analysis of the number of vulnerabilities in a protocol 26

Table 4.3: Number of packets sorted per protocol 26

Table 4.4: Process of the vulnerability score 27

Table 4.5: Weight calculation 28

Table 4.6: Records of attacks for an appliance 28

Table 4.7: Example of the Protocol protection matrix 28

Table 4.8: Calculation of the security index 29

Table 4.9: Example of the new vulnerability score 30

Table 5.1: Description of servers and their purpose inside the network 35

Table 5.2: Installation steps for Netkit 39

Table 5.3: Configuration example of DNS Server 41

Table 5.3 (Continued): Configuration example of DNS Server 42

Table 5.4: Configuration of the Web server 43

Table 5.5: Procedure to generate random traffic 45

Table 5.5 (Continued): Procedure to generate random traffic 46

Table 5.6: Number of packets for each protocol 47

Table 5.7: Amount of vulnerabilities in HTTP 47

Table 5.8: Amount of vulnerabilities in DNS 48

Table 5.9: Amount of vulnerabilities in ARP 48

Table 5.10: Amount of vulnerabilities in ICMP 48

Table 5.11: Vulnerability score for each of vulnerabilities 49

Table 5.12: Weights for each protocol in the experiment 49

Table 5.13: Protocol protection matrix 49
Table 5.14: Security Index 50
Table 5.15: Amount of vulnerabilities in HTTPS 50
Table 5.16: Change in the Vulnerability score by replacing HTTP with HTTPS 50
Table 5.17: Change in the Vulnerability score by filtering ICMP 51
Table 5.18: Protocol protection matrix by using IPS 51
Table 5.19: New security index by applying the IPS 52
Table 5.20: Protocol protection matrix by using an application firewall 52
Table 5.21: New security Index by applying the application firewall 52
Table 6.1: Comparison of the security index scores 54
Table 6.2: Comparison of the vulnerability scores 55
Table 6.3: Threat indices for the experiment 58
Table 6.4: Categories for the upgrades 58
Table 6.5: Risks covered by the technologies 59
Table 6.6: Effective estimates 59
Table 6.7: New Threat Indices 60
Table 6.8: Comparison of the results 61

Acronyms

ARP: Address Resolution Protocol

CAESARS: Continuous Asset Evaluation, Situational Awareness and Risk Scoring

CVSS: Common Vulnerability Scoring System

DNS: Domain Name System

IP: Internet Protocol

LAN: Local Area Network

MTTF: Mean Time to Failure

NIST: National Institute of Standards and Technology

NVD: National Vulnerability Database

SA: Situation Awareness

SAEM: Security Attribute Evaluation Method

TCP: Transmission Control Protocol

VPN: Virtual Private Network

1 Background

1.1 Introduction

The current trend for a rapid connectivity system that enables quick responses within an organization and complies with the essential requisites for interconnecting the employees via a network is an essential need within an enterprise. Nonetheless, enhancing technological assets for an enterprise is a time consuming and complex task which involves an understanding of what the business is, what it already has, the needed requirements for improving its present situation, and what it can afford.

In that sense, to protect an enterprise network and the elements which embody it (computers, servers, switches, software appliances, etc), most companies allocate a budget in acquiring security appliances (firewalls, antiviruses, etc) that gives an important layer of security to protect critical operations.

This scenario generates the need of deciding the most appropriate security solution according to the company's needs. Not addressing this issue properly could lead to a vulnerable environment and unnecessary acquisitions. One example is how the ransomware WannaCry affected several computers and servers because companies have not installed a software upgrade for Windows Systems on their computers and servers (Symantec Corporation, 2017). While some companies could have relied on antivirus software or a more complex solution, an analysis of the network could have led to the need of an immediate upgrade on Windows operative systems, avoiding the outcome of the infection.

There are currently different options for analysing threats and vulnerabilities inside a network. The Common Vulnerability Scoring System (CVSS) is an example of how vulnerabilities can be scored and therefore compared. Furthermore, there are currently papers and studies about how many resources should be destined for protecting the network and how to address possible alternatives (Fielder, et al., 2016; Butler, 2002). However, as technologies and systems keep changing, previous studies should be

reviewed an updated accordingly.

The focus of this project is to develop a security metric-based model for measuring different security solution alternatives to help in the decision-making process of choosing a security solution and increase the overall security. The model should be able to contrast the level of security with and without upgrades, giving an opportunity for decision-making people to take more accurate choices in relation to how to invest in cybersecurity for a company. The last part of the research will involve testing the developed model as part of an experiment and compare the results with other models.

The project will comprehend communication protocols inside the network, regardless of the applications and operative system inside those elements. It will focus on the application protocols travelling through TCP/IP networks and assess the level of security according to the vulnerabilities inside those protocols as well as how much the proposed upgrades can increase the security level. Software vulnerabilities related to operative systems or programs inside the terminals would not be part of the scope because of the time constraints and resources available. However, the model will be open for introducing other metrics based on software vulnerabilities as part of future work.

1.2 Research question

This research answers the following research question:

In what way the threat level and the cost-benefit of a possible security solution oriented to a TCP/IP network can be analysed

To answer this question, the following objectives have been identified:

1. Critically analyse security metrics and models and their benefits in revealing the overall network security for a company
2. Develop a conceptual model which provides information aiding in the decision-making process of acquiring a security solution
3. Test the proposed model in a case study to validate the resulting metrics

usefulness

4. Analyse the results of the test and propose guidelines on the use of the model

1.3 Dissertation structure

The process to fulfil the objectives in answering the research questions are shown in Figure 1.1. Starting with Chapter 3, a literature review will take place to state the current level of research regarding metrics and models as stated in Objective 1. This covers the analysis of papers and studies around metrics, and the models used to understand them. The findings will reveal models used to evaluate security in a network.

Next, the findings will be used to determine the most suitable steps and metrics needed for the model. Chapter 4 presents an analysis to determine a suitable model to answer the research question. This new model will consider: the information needed as an input, the processes to understand and analyse the information, and the expected type of results.

The discussion continues in Chapter 5 were a case study will test the model and provide the results according to what was found. The analysis of the collected metrics will compare the actual state of the network with different upgrade proposals and determine the most suitable one according to the model.

Finally, the analysis in Chapter 6 of the results and how they contrast with other models will allow proposing the guidelines of how to apply the model based on the experience obtained in the previous chapter.

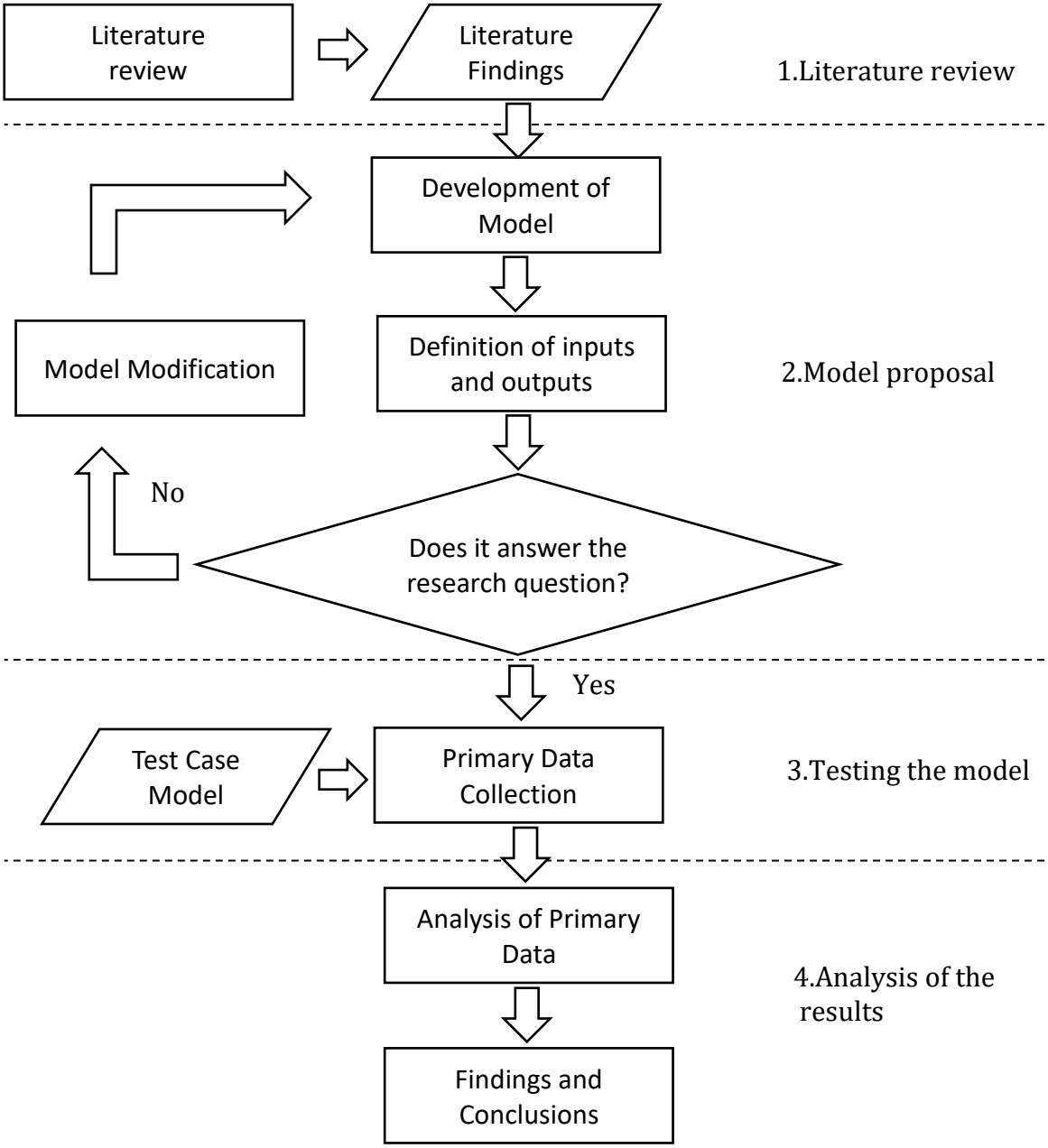


Figure 1.1: Dissertation's structure

1.4 Motivation

Currently, there are different approaches to address measuring the level of security inside a network in real time, but not oriented to determine possible upgrades. IT managers define the level of security based on their experience and knowledge of the network, making it difficult to measure a possible solution when it has not been implemented yet.

Furthermore, even if the engineers are capable of measuring the threats in a networking environment and determine the benefits of a security upgrade, it is difficult for them to quantify how much a security solution would improve the current state of the network. By using a model to quantify the impact of a solution for everyone, enterprises could measure their investment and avoid acquiring over-dimensioned solutions.

A quantifiable level of security in a digital environment is important for engineers. They will try to increase the security making the best use of the technologies at hand, but those technologies (appliances or software) have limits. By having a model with different ratings, people with or without a technical background of cybersecurity could answer their concerns about security and cost-benefit analysis for investing in technological solutions.

2 Research Methods

In the previous chapter, the research question was stated as well as the strategy to answer the question. Before evaluating previous studies around network security, the analysis procedure is explained in the following lines.

This chapter aims to justify the different approaches taken in consideration for answering the research question. It states the justifications for the methods applied and the foundation for the analysis of data. The aim of the research will be presented as well as the objectives. For successfully achieving each objective, the different options to do so are shown as well as the decision-making process for selecting the most suitable option.

The research philosophy and approach explain further the reasoning behind the resulting arguments made in the study. Finally, the design and strategies provide more insight into how the data will be treated to produce an outcome that will meet the objectives set.

2.1 Research aim

This research was conducted to determine a model for analysing the level of security in a network. The scope will be centred on Local Area Network (LAN) environments using the TCP/IP model for communication. In order to achieve the aim of the study, four research objectives have been proposed in Section 1.2:

1. Critically analyse security metrics and models and their benefits in revealing the overall network security for a company
2. Develop a conceptual model which provides information aiding in the decision-making process of acquiring a security solution
3. Test the proposed model in a case study to validate the resulting metrics usefulness
4. Analyse the results of the test and propose guidelines on the use of the model

Sections 2.1.1 to 2.1.4 elaborate in detail each objective and presents the strategy for

achieving an answering these objectives.

2.1.1 Analysis of metrics and models

In Objective 1, the term metric is defined by taking in consideration different studies related to security, and how it addresses relevant information from the network to be aware of the internal level of security. However, a metric alone would not give an overall vision of the security inside a network. For that reason, metrics should be processed and correlated to simplify the decision-making analysis without leaving behind important parameters about the network's security. That process is described as a model.

This chapter will set the foundation for the study, as it should define the concept and features of metrics and models. Furthermore, metrics with similar characteristics will be grouped into categories for a better comprehension of all the possible metrics a network can provide. Finally, a comparison of models will deliver the mechanisms of how the metrics can be processed to give data related to the network's security.

There are two possible research methods to fulfil the purpose of the chapter. In order to give a proper definition to the word metric, the first method is performing a survey to security or network analysts that work directly with metrics and models and making a list of metrics and models used by them. However, contrasting the different answers could prove to be time-consuming, and their answers would not necessarily cover all the possible metrics.

The other method involves using previous research. They can be more rigorous in their findings as they have to explain how they reach those conclusions. A critical analysis will result in finding other researcher's definition for metrics and models and proposing a definition according to the scope. This option can provide even more information than a survey and would allow gathering more data if needed. In conclusion, analysing previous research related to the subject will be chosen.

2.1.2 Model development

Objective 2 opens the possibility of either modifying an existing model or developing a

new one for helping in the decision-making process. The first approach considers the information obtained in the previous objective by taking the metrics and processes from other models. Even if the literature and analysis could provide the most suitable model, it will be very likely that it would not fulfil entirely the research's question. However, with modifications over an existing model, the outcome would be a new model adapted to the requirements of the objective and with previous research to support it.

Another possibility is establishing a new model based on the different processes used by the other models. The risk of that approach is that designing a model involves a deeper analysis of mathematical formulas and further discussion that would exceed the time constraints for the research.

By modifying previously defined models, the inputs and outputs needed to provide an answer are already defined and considered. The inputs will include the metric's categories found previously, while the outputs will be determined based on what the chosen model can provide. Finally, the inputs, outputs and processes inside the model will determine the structure of the new model.

2.1.3 Testing of the model

Since the research is introducing a new model, it will require to be tested and the results it provides to be validated. As the inputs require metrics taken from a network, the test will require implementing a set of network elements (real or virtual appliances) where the metrics will be measured. Each element will use a set of communication protocols to provide a service or to interact with other elements.

As mentioned, the test can be performed by two methods: using a real or virtual environment. Creating an environment with real equipment will require additional resources (servers, switches, router, etc) not available because of the budget and time constraints of the research. Furthermore, it will be difficult to predict the behaviour of a real network.

On the other hand, generating a virtual environment may not be able to replicate completely the network's behaviour as resources are limited to emulate an entire

network. However, a virtual environment allows more control over the configuration inside each element and over the behaviour of the network, and because of that, this option will be used to recreate the network where the model will be tested.

By using a virtual network, this objective will present a case study with an analysis of network elements to be considered, and the steps to reproduce the experiment. After the network is deployed and security upgrades are proposed, the model will be used to address the level of security and how beneficial the upgrades are. As the possible upgrades are compared to determine the most beneficial solution, all the results and outcomes will be summarized for further analysis.

2.1.4 Analysis of the results

In the last part of the research, the project will aim to validate the results of the case study to determine if it really reflects the security and the benefits of the upgrades. Because a new model is being proposed, an option for validation will be comparing the results with the literature of other models and contrasting them. Alternatively, another option is making interviews with IT/Security managers who address the security needs of the networks they look after. They can give their views about the new model's results and compare them with their own experience in analysing and proposing solutions inside their companies.

Both options would generate the necessary information about the validity of the research. However, since organizing interviews would not necessarily provide with accurate information and performing a survey on a significant sample of people could take too much time, another option should take place. That is why comparing the results with other models is chosen as the option to follow.

Finally, in Appendix A and B, both the Consent Form and the Participant Information Leaflet (PIL) are presented. In Appendix C, the Ethical Approval confirmation is attached.

2.2 Research philosophy

When analysing a network, values like the number of attacks received or Mean Time to Failure (MTTF) of a server can be easily measured and compared, while the description of a vulnerability or the level of risk as the result of applying a model inside the network can be subjective and open for interpretation. Because the network will produce both types of data, qualitative and quantitative metrics, the research will adopt the pragmatism philosophy (Wilson, 2014), as it covers the importance of both metrics.

2.3 Research approach

As mentioned in the objectives and the research philosophy, some metrics are open to interpretation. When detecting a port scan or a forbidden connection to a server, they could be part of a bigger attack path. Analysing the data will require making inferences for a possible model that will answer the research question and satisfy the objectives goals.

The research will also consider pragmatic considerations, as comparisons for reaching the best processes for a model are part of the second objective. A deductive approach would not properly analyse subjective metrics. They could have different meanings for each researcher, and the inductive approach will look to generalize the result. The approach should orient the study to the closest explanation of what the research question raises.

For that reason, the study orients the research to an abduction approach (Josephson & Josephson, 1994): the research will present a model that can answer the research question better than the other models or alternatives.

2.4 Research Design and Strategies

Since the options to answer each objective have been selected, it should be mentioned the type of literature review to be adopted for the objectives. It will be stated the sources and

the literature review method to be used.

2.4.1 Literature sources

The key themes to be addressed in this research are:

1. Different methods to analyse the level of security in a network
2. Strategies to determine the most suitable security solution

For the objectives, qualitative and quantitative data from journal articles, conference papers, books and standards will be gathered. Previous studies will cover the themes of the research and establish the foundation for the virtual network for analysis.

2.4.2 Literature review method

The traditional and systematic literature review are possible options for the method to be used in the research because a meta-analysis would not be able to analyse the qualitative metrics, and meta-synthesis will not handle quantitative data from the network. From both possible options, the traditional literature review can identify gaps between the models as it analyses and summarises all the data gathered, while the systematic literature review evaluates it in a more rigorous approach according to the level of complexity of the question (Arshed & Danson, 2015).

Currently, there is a lot of research related to address the level of security inside a network and how to calculate metrics for network security. However, the complexity of the question is around how to measure the level of security that a possible security solution will provide. That is why the systematic literature review approach will be used for understanding the data and approaching a solution.

2.5 Conclusion

This chapter summarized the methodology behind the present study. It presents the analysis of the choices made to examine the information and fulfils each objective. The pragmatism philosophy and the abduction approach are the basis of answering the

research question, while the systematic literature review will be followed for understanding the information gathered.

3 Literature review

In the background, the need for a model to analyse the threat and the cost-benefit of a solution is stated as the main goal of the research. For achieving that purpose, the literature review will focus on previous research about the information provided by the network to obtain a level of understanding of how secure the network is.

Furthermore, the information will set the definition of security metric, and the values related to them will be sorted into categories for analysing how they can be measured and treated.

After that, the study will define models currently used for decision making regarding cybersecurity, comparing them in terms of the inputs needed and the evidence they provide for helping in the decision-making process. This will cover previous research regarding assessing the level of security: books, journal articles and conference papers. Since these sources are supported by experiments or theoretical analysis, they can be critically analysed and compared with each other.

3.1 Security metrics

To assess the level of security of a network, it is imperative to define what is situation awareness (SA) and how it is applied to the model to be built. Situation awareness is defined as "the perception of the elements of the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future" (Tadda & Salerno, 2010). The elements of the environment in a network are the data or evidence provided by the network; the time and space are defined by the current state of the network, and the projection would be the possible security solutions to be installed for upgrading the security of the network.

Regarding defining metrics, considering metric solely as data or evidence in the network is a wide term. The elements to be considered as metrics are the evidence that provides information about the performance of the network in terms of security. For example, logs

generated by switches, servers, routers, firewalls, and other appliances inside the network are key sources of data, with the capacity to be organized in levels (events, alerts, etc) (Nathans, 2015). Logs provide a great amount of information, but not always related to security.

Nathans (2015) also mentions incidents as a violation or imminent threat of violation of security policies that can be translated into denial of service, data loss, and more. An example of that is a virus, not recognized by the antivirus software, propagating through the network without leaving logs but creating other types of evidence like unknown traffic being sent by some machines or denial of service as an outcome. From the analysis, it raises the idea that logs are not the only source of information a metric should consider.

Li et al. (2010) remark how metrics are used in the Common Vulnerability Scoring System (CVSS) as elements with the ability to describe and measure the properties of a vulnerability. This statement complements the first one as it not only focuses on the path of the attack but the severity too. CVSS scores are numerical values to indicate the severity of a vulnerability allowing to assign a level of severity. Similarly, Mateski, et al. (2012) describe metric as *'a consistent standard of measurement'* but making clear that *'metric is a unit of measure'*: when we measure an attribute or particular behaviour (metric), it will give us a number (measurement). A well-defined metric helps to understand the attribute that is being analysed. By obtaining a value, a metric can standardize the attributes and help in improving the system or comparing it to a previous state.

Finally, by looking into Pendleton et al. (2016), security metrics can reflect an *'attackers attempt to exploit system vulnerabilities'*. The word *'attempt'* adds unsuccessful attacks to the definition as they could be considered in the decision process. It also defines metric as a value and not the measuring process. From all these statements, a security metric can be defined as an assigned value product of a standardized measurement of any element inside the network with the capacity to describe attack attempts, exploitable vulnerabilities, or the path of the attack to any element inside the domain.

3.1.1 Categories of metrics

As mentioned before, it is not possible to foresee all the possible variables inside a system.

Nevertheless, information from the network can be grouped based on different attributes. One example is grouping metrics based on the level of abstraction (Barford, et al., 2010):

- **Low-level data:** Variables concerning vulnerability analysis, attack correlation, information flow analysis, etc. These metrics alone are not always enough for the security analysis and they require the insight of the security manager to determine a possible attack. Moreover, the quantity of this kind of data can become overwhelming, for example having thousands of logs to analyse.
- **High-level data:** The manager's analysis of previous attacks, the ability to correlate incidents and to translate variables into complex attacks is an important factor. Even if this labour of evaluating these metrics is done manually and is prone to human error, it cannot be left behind.

On the other hand, Pendleton, et al. (2016) identifies a set of metrics to determine the level of security and how to measure them, based on previously calculated mathematical functions. A summary of the metrics mentioned in their research is in Appendix D.

Pendleton, et al. (2016) recognizes how security is affected by four vectors: attack, defence, vulnerability and situation. A factor like CVSS that was used by Li, et al. (2010) to explain how this scoring system can address the level of vulnerability and is used to define a security metric, is little compared to the big amount of metrics a system can provide not only in terms of raw data but in the knowledge an IT manager can provide about the network (for example situation metrics).

Another possibility is categorizing each metric according to its dimensions: confidence, purity, cost utility, and timeliness (Tadda & Salerno, 2010). More details about these dimensions are described in Appendix E.

Their research provides with mathematical formulas that calculate some of the metrics mentioned inside it. Parameters like confidence and purity add a new layer to consider, since Pendleton, et al. (2016) do not consider how reliable the information can be. However, assessing the metrics mentioned can only be applied in networks where data has enough level of detail to differentiate a single attack from a more complex group of attacks working together for the same purpose.

One final approach to consider is how to visualize the metrics. Duggan, et al. (2007) presents not only categories of metrics but how to present them. A brief description of their work is presented in Appendix E.

The definition of their metrics is too generic to be considered for security metrics and seems to be more suitable to describe a physical attack or security incidents outside the field of the TCP/IP network. Nevertheless, focusing on the relevance of visualization is important, as this research provides a matrix to compare threats and could be used in the following sections for presenting the data.

Finally, some of the metric categories mentioned are part of a continuous risk assessment to maintain and improve the level of security of the system in general and some of them are focused on software. When defining the model to be used for answering the research question, some metrics will not be considered as they are not part of the scope of the research question.

3.2 Models to assess the level of security

Over the years, several studies have been released to answer the need of measuring the threat level for organizations. With the great number of metrics gathered into categories, the following steps involve defining a process to simplify the analysis.

3.2.1 The Cyber Prep methodology model

Bodeau, et al. (2010) present a simple scheme to present threats, indicating that the structure they tested can be applied to most of the organizations. The inputs of their model are three metrics oriented to the attacker threatening the organization: Capability, Intent and Targeting. Each metric has values between 1 to 5 whereas 5 is an advanced or critical threat to consider. For each input, every level has a statement to help in allocating the threat into the correct value.

Threat Level	Capability	Intent	Targeting
5: Advanced	The adversary is very sophisticated and well resourced and can generate its own opportunities to support multiple successful, continuous, and coordinated attacks.	The adversary seeks with great determination to undermine, impede severely, or destroy, a mission, program, or enterprise, by exploiting a presence in the organization's systems or infrastructure. The adversary is concerned about disclosure of tradecraft only to the extent that it would impede their ability to complete their goal.	The adversary analyzes information obtained via reconnaissance and attacks to target persistently a specific organization, enterprise, program, or mission, focusing on specific high value or mission-critical information, resources, supply flows, or functions; specific employees or positions; and supporting infrastructure providers and suppliers and on partnering organizations.
4: Significant	The adversary has a sophisticated level of expertise, with significant resources and opportunities to support multiple successful coordinated attacks.	The adversary seeks with determination to undermine or impede critical aspects of a mission, program, or enterprise, or place itself in a position to do so in the future, by maintaining a presence in the organization's systems or infrastructure. The adversary is very concerned about minimizing detection of their attacks or disclosure of tradecraft, particularly while preparing for future attacks.	The adversary analyzes information obtained via reconnaissance to target persistently a specific organization, enterprise, program, or mission, focusing on specific high value or mission-critical information, resources, supply flows, or functions, specific employees supporting those functions, and/or key positions.
3: Moderate	The adversary has moderate resources, expertise, and opportunities to support multiple successful attacks.	The adversary seeks to obtain or modify specific, critical information and/or to usurp or disrupt the organization's cyber resources by establishing a foothold in the organization's systems or infrastructure, but is concerned about minimizing detection of their attacks or disclosure of tradecraft, particularly when carrying out attacks (e.g., exfiltration) over long time periods. The adversary is willing to knowingly impede aspects of the organization's mission to achieve these ends.	The adversary analyzes publicly available information to target persistently specific high value organizations (and key positions, such as Chief Information Officer), programs, or information.
2: Limited	The adversary has limited resources, expertise, and opportunities to support a successful attack.	The adversary actively seeks to obtain critical information and/or to usurp or disrupt the organization's cyber resource, and does so without concern about detection of their attacks or disclosure of tradecraft.	The adversary uses publicly available information to target a class of high value organizations and/or information, and seeks targets of opportunity within that class.
1: Unsophisticated	The adversary has very limited resources, expertise, and opportunities to support a successful attack.	The adversary seeks to usurp, disrupt, or deface the organization's cyber resources, and does so without concern about detection of their attacks or disclosure of tradecraft.	The adversary may or may not target any specific organization or class of organization.

Figure 3.1: Adversary Levels Table (Bodeau, et al., 2010)

Each organization could allocate their threat sources (adversaries) according to the organization's needs. The process starts by assigning the corresponding threat value to the Capability, Intent and Targeting input by matching the adversary's description with

the corresponding level. However, there is no specific way to put a weight on the overall threat level: one company could use the upper value of the three while another can use the average or the lowest one as the level of threat of the adversary.

3.2.2 Attack criticality assessment framework

Raulerson, et al. (2015) presents another model aiming to assign a value of severity to attacks performed on a network from a scale of 1 (immune) to 7 (disaster) with the purpose of providing situation awareness. The value assigned to each element inside the network corresponds to the level of threat of the attack according to CVSS. However, its purpose is only to sort out the amounts of data.

For testing the model, a network was built and a total of 699 cyber-attacks were performed by using a tool called *BackTrack5*. The results enabled the person in charge to know the impact of the attacks and allow him or her to focus on the data that the person finds relevant. The test included all the attacks whether they were or not successful, implying that the model does not asses the defences in place.

3.2.3 CAESARS Framework

The National Institute of Standards and Technology (NIST) defines another model known as Continuous Asset Evaluation, Situational Awareness and Risk Scoring (CAESARS) that relies on continuous monitoring of a network by implementing a system consisting on four modules: Sensor, Database, Presentation/Reporting, and Analysis/Risk (Mell, et al., 2012). While the Database is used to save data from the sensors, the other modules work with different tools to perform their purpose and remain independent from each other.

- **Sensor:** It has nine sensor types in total and performs a continuous analysis of the network. It takes vulnerabilities, network configuration, and others as data inputs.
- **Database:** Repository of raw data and recordings from both sensors and auditing tools. It is also linked to other databases, like the National Vulnerability Database (NVD)
- **Analysis/Risk Scoring:** It contains tools for analysing the information in the database.
- **Presentation/Reporting:** Include different types of reports, providing several ways

to display data.

Among the limitations of CAESARS model, a critical limitation is the lack of an established way to measure the security risk. It does not explain how the raw data is transformed into a Risk Score to determine the effectiveness of the defences in the network. More than a model to assess the level of security, it describes a set of elements needed to deploy a monitoring system in the network. While some organizations could lack on some of CAESAR's elements, it is worth mentioning that they rely on a vulnerability database to stay informed of known threats and how important are the scoring and proper presentation of the results.

3.2.4 SAEM analysis model

In another study, Butler (2002) produces a case study where a Security Attribute Evaluation Method (SAEM) takes a risk assessment as an input and produces a cost analysis to estimate the necessary security investment needed. More details about this model are explained in Appendix F.

This model provides a complete view of how to decide from a list of security options the one that most suits them. The outputs could be well understood by non-IT professionals, as the normalized effectiveness, security coverage and costs provide with different perspectives to see and consider. However, several input metrics are subjective and depend on the network or security manager.

In summary, each model presented suits a particular purpose and considers only the metrics relevant to achieving their goal. Even with different purposes, they follow a common process in doing so: receive a quantity of data input, analyse and process the data, and deliver an output that simplifies the understanding of the security in the network (or a particular technology upgrade).

Inside the process of analysis, the models manage to normalize the inputs via mathematical algorithms or a subjective ranking process. However, the constant use of subjective metrics can alter easily the results of each model, depending on the person making the analysis. For proposing a model, this last observation should be considered.

4 Model proposal

From the literature review, several studies provided an insight into the metrics and models that can be used in network security. In this chapter, a discussion will be developed to further analyse each of the discussed metric groups and models. Henceforth, the conclusions of the discussion will show the metrics that this model will consider for analysis and their outcomes. Once the metrics and outcomes are verified, then the process to handle them will be stated as the model proposal.

4.1 Model comparison

By taking into consideration what is involved in the research question, what the proposed model should consider must include:

a) The identification of available metrics to denote the current security of the network (Zhai & Wang, 2011)
b) An estimation of the current situation regarding security
c) Determine the new status of the network in terms of security after applying one or more secure solutions.

For the first point, the research from Pendleton, et al. (2016) provided an enriched set of metric categories which will be used in detail to clarify the focus of the previous models. In Table 4.1, the highlighted words in the table are the metrics names used by each model already mentioned. Below these highlighted metrics, the table shows other metrics used by Pendleton, et al. (2016) to support the discussion.

A first look into the table shows that, apart from the state of the network, SAEM analysis also considers the security upgrades and their cost as part of the model categories. Thus, each model focuses on a particular goal.

Table 4.1: Comparison of metrics used by different models

Model	Input metrics	Output metrics
Cyber Prep methodology	**Capability:** a) Attack Evasion **Intent:** b) The severity of Individual Software Vulnerabilities: **Targeting:** c) Targeted attacks	**Threat level:** d) Security State
Attack criticality assessment framework	**Records of attacks performed to the network:** e) Security Incidents **Vulnerability Database:** f) The severity of Individual Software Vulnerabilities	**Activity of Interest** g) Security State
CAESARS Framework	**Sensors** h) Attack, Defence, Vulnerabilities and Situational metrics could fit in this area **Vulnerability Database** i) The severity of Individual Software Vulnerabilities **Previous Findings** j) Security Incidents	**Situational Awareness** k) Security State
SAEM analysis	**Outcome attributes** l) Security Incidents **Alternative Effectiveness, costs and coverage** m) Various Defence metrics n) Security investment	**Security State** o) Security State **Cost-Benefit analysis** p) Security Investment

The Cyber Prep methodology consider mostly attack metrics since their main concern is the attacker. The attack criticality assessment framework is more concerned about how to appropriately summarize an attack's information. On the other hand, CAESARS Framework does a complete assessment based on several sets of metrics.

SAEM analysis has a different approach than the others. In this paper, the metrics are mostly related to previous attacks, and it also relays on subjective data that is the enterprise's concern about certain outcomes. In the end, the outputs are the security state and also a comparison between possible upgrades.

Most of the models in Table 4.1 are more focused on attack metrics than the current security installed in an enterprise. The Cyber Prep methodology and the Attack criticality assessment framework indirectly consider defence metrics by considering the capability of a threat inside a network, or how many successful attacks were made. SAEM analysis only focuses on potential outcomes. Only when looking for security alternatives, some defence metrics are taken into consideration by SAEM analysis. CAESARS Framework collect several pieces of information that include metrics provided by security appliances or solutions but do not specify the analysis for understanding the level of security based on those metrics.

By considering the different categories of metrics mentioned by Pendleton, et al. (2016), a similar approach for that option can be used with SAEM analysis. SAEM currently considers only past security incidents inside the Situation category. If the same can be done for other categories mentioned in Pendleton's work, the current defence and other variables could be part of the security assessment.

4.2 Establishing the inputs and outputs needed

For proposing what a model should consider, it is needed to add the definition of input and output for the model. An input will be any information provided by the system that will help in the pursuit of assessing the current state of the network in terms of security. This definition includes metrics and other elements related to the state of the network. The output should be one or more metrics, as well as any other values that can help in the decision-making process of implementing a security solution.

After some analysis of the information inside the network and the expected outcome of the model, and by taking as reference the SAEM analysis model, Figure 4.1 presents a diagram of the summary of inputs and outputs for the model.

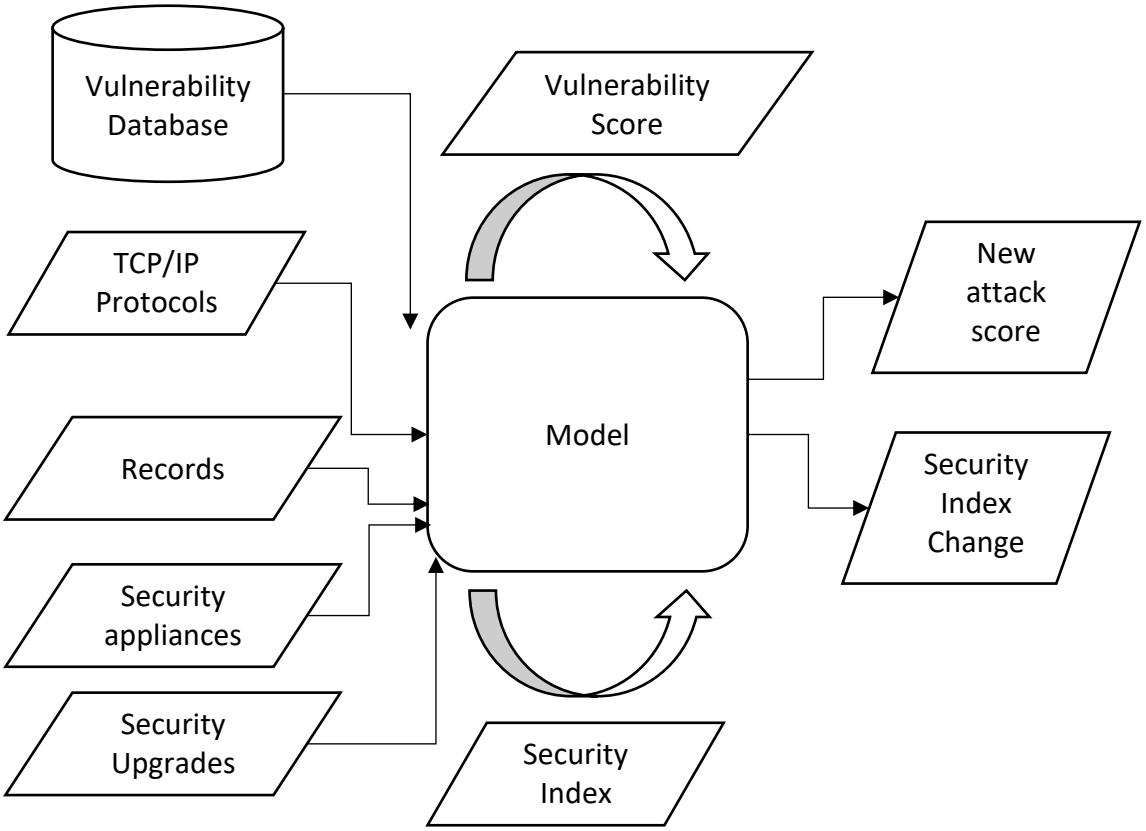


Figure 4.1: Diagram of inputs and outputs for the model

4.2.1 Analysis of the inputs

The following explains in detail the inputs that the research question requires to consider:

a) **TCP/IP Protocols:** The model should consider the communication protocols inside the network. This includes values as *"number of packets"* and *"connections associated to each protocol"*.
b) **Security upgrades:** It involves the upgrades to be proposed and compared. Each upgrade is associated with the protocols they protect.

Additionally, the following inputs are needed:

a) **Vulnerability Database:** It involves an external database gathering reported vulnerabilities and analysing them. The reason to include it is that the model will rely on the metrics provided by this source to assess the vulnerabilities associated with each protocol.

b) **Records:** Depending on the elements inside the network, previous information like *"number of successful attacks"* can show the status of the network. However, the model should also consider some cases were logs and other data are not stored, so no record is present for the analysis.
c) **Security appliances:** Any appliance currently being used to protect the network from attacks is part of the analysis. It also describes the status of the network.

4.2.2 Analysis of the outputs

Some of the results of the model should be used to compare the current state with the future state of the network. Those metrics are mentioned next:

a) **Vulnerability Scores:** The model will produce a set of scores to determine the current threat level of the system. This first output will be used as an input to compare how much it changes when a new upgrade is proposed.
b) **Security Index:** This value scores the level of security in the network. It alone has no meaning until a new solution is proposed and the value changes.

However, as seen in the SAEM analysis, the score by itself is not enough. They only set a reference to the current state of the network. That is why the output values are defined as the following:

a) **New vulnerability scores:** It is the result of analysing the change between the initial and the new attack score value. It compares the result of replacing or disabling a protocol.
b) **Security index change:** This value is the result of comparing the new security index with the previous one. This value can be compared with the result of applying other upgrades.

Additionally, there is work related to security budget calculation and cost analysis of possible upgrades that could be added. However, it is more important to determine how beneficial a solution is before considering costs. The difference in costs could be a final metric only in case there are two or more upgrades with a similar security index and vulnerability score. For that purpose, each company can rely on their own cost assessment

to differentiate two or more upgrades with similar scores.

4.3 Establishing the model to be proposed

After comparing and analysing the models in Section 4.1, and by using SAEM analysis as a reference, the steps for the model are as follows:

4.3.1 Establishing the monitoring point

The first issue to consider is the asset to protect. That asset could be a group of users or servers which the company considers of great value. The traffic going from or to those assets will show the protocols to consider.

After defining the assets, the next step is to establish the monitoring point. The monitoring point should be the point closest to the assets where traffic can be collected. This point will define the security appliances to consider in the analysis; in other words, the appliances meant to protect the traffic of interest that passes through them.

The last step is gathering the records of the traffic of those assets. If the network has a system for gathering and storing traffic logs and attack logs, then that solution can be used for the analysis. Otherwise, it will be essential to monitor the network and gather enough traffic to describe the behaviour of the network. While a minimum of traffic has not been established for saying how much it represents the normal traffic inside a network., it will be the task of the network manager to gather the necessary data for that purpose.

4.3.2 Establishing the Vulnerability Score

The vulnerability score considers as inputs the information from the Vulnerability Database and the records of the number of packets for each TCP/IP protocol. The Vulnerability Database will present all the vulnerabilities related to each protocol found in the monitoring point. The records will indicate the amount of traffic for each protocol that passes through the network.

After gathering the vulnerabilities associated per protocol, depending on the score, they will be sorted in ranges of how critical they are. If the Common Vulnerability Scoring

System (CVSS) is used for scoring the vulnerabilities, the ranges are already defined by its documentation. The values in CVSS go from 0 to 10 and depending on the value of the vulnerability, its sorted in a range: Low, Medium, High and Critical (FIRST.org, 2018).

An example is shown in Table 4.2 where two protocols are considered:

Table 4.2: Analysis of the number of vulnerabilities in a protocol

Range	Number of Vulnerabilities	Percentage
Low (0.1-3.9)	a	a/E
Medium (4-6.9)	b	b/E
High (7-8.9)	c	c/E
Critical (9-10)	d	d/E
TOTAL	$E = a + b + c + d$	100%

In summary, the formula to express the values in each range is the following:

$$Range\ X\ percentage = \frac{Number\ of\ vulnerabilities\ in\ Range\ X}{Total\ number\ of\ vulnerabilities}$$

Figure 4.2: Formula to calculate the percentage of vulnerabilities in each range

After doing the procedure in Table4.2 for each protocol, the next step is sorting the records of traffic for the protocols. Table 4.3 is an example of the expected outcome, were M and N are the number of packets for each protocol.

Table 4.3: Number of packets sorted per protocol

	Protocol 1	Protocol 2	TOTAL
Number of packets	M	N	$M + N$

The final score is the product of the number of packets and the percentage of packets of each protocol. This method is similar to the Threat Index calculation described by the SAEM analysis in terms of normalizing values. For Table 4.4, the variables presented in Table 4.2 and Table 4.3 are used as a reference. For Table 4.2, *"a1"* and *"a2"* are different to express their belonging to Protocol 1 and 2. This applies to values *'b'*, *'c'* and *'d'*.

Table 4.4: Process of the vulnerability score

	Protocol 1		Protocol 2		Score
	Percentage of vulnerability score	Percentage of packets	Percentage of vulnerability score	Percentage of packets	
Low	$A1 = a1/E$	$L1 = \frac{M}{M+N}$	$A2 = a2/E$	$L2 = \frac{N}{M+N}$	$(\boldsymbol{A1} \times \boldsymbol{L1}) + (\boldsymbol{A2} \times \boldsymbol{L2})$
Medium	$B1 = b1/E$		$B2 = b2/E$		$(\boldsymbol{B1} \times \boldsymbol{L1}) + (\boldsymbol{B2} \times \boldsymbol{L2})$
High	$C1 = c1/E$		$C2 = c2/E$		$(\boldsymbol{C1} \times \boldsymbol{L1}) + (\boldsymbol{C2} \times \boldsymbol{L2})$
Critical	$D1 = d1/E$		$D2 = d2/E$		$(\boldsymbol{D1} \times \boldsymbol{L1}) + (\boldsymbol{D2} \times \boldsymbol{L2})$

The last column presents the first output the model provides. Each protocol will influence depending on the number of packets of that protocol and its score. For example, in case there is a protocol with a large number of critical vulnerabilities but with only a really small quantity of packets compared to the rest, this protocol would not greatly influence the result.

4.3.3 Establishing the Security Index

The security index considers the following inputs: the number of packets for each TCP/IP protocol, number of attacks registered, and the characteristics of each security appliance. For the last input, the relevant characteristics are the protocols the security appliance can protect.

Another input that has not being discussed is how essential is a service. For example, elements like a web server could be relevant for a company. By using the Swing Weight Method (Butler, 2002), instead of attributes, each protocol will be set with a rank of 1 to 100. The rank will be set up by the manager according to the critical services of the company and the protocols used by those services. There is no standard of how to set a rank, but the statements used to rank each system should be consistent for each protocol. The only purpose is to reflect the concern of the managers regarding the priority to protect some protocols. An example of the ranks is presented in Table 4.5.

Table 4.5: Weight calculation

	Protocol 1	Protocol 2
Rank (1-100)	f	g
Weight	$F = \frac{f}{f+g}$	$G = \frac{g}{f+g}$

The next step is the analysis of the records of previous attacks. For each security appliance in the network, there will be registered attacks and probably some of them were successful ones. In case there are no records, a penetration test could be performed to set a finite number of attacks associated with each protocol and register how many where successful. In case that option is not possible, the model can assume that the security appliance is working flawlessly.

Table 4.6: Records of attacks for an appliance

Appliance 1	Protocol 1	Protocol 2
Number of successful attacks	$p1$	$q1$
Total number of attacks	$p2$	$q2$
TOTAL	$P = \frac{p1}{p2}$	$Q = \frac{q1}{q2}$

Afterwards, it is necessary to consider which appliances can protect each protocol. For example, appliance 1 has the capacity to protect protocol 2 but no protocol 1. However, appliance 2 can protect both. The result of this analysis is a matrix as shown in Table 4.7. A value of '1' is present when the appliance can analyse a protocol. Otherwise, the value is '0'.

Table 4.7: Example of the Protocol protection matrix

	Protocol 1	Protocol 2	…
Appliance 1	0	1	…
Appliance 2	1	1	…
…	…	…	…

Finally, by using the dimensionless units in each step it is possible to capture the relevance of each element in terms of security. By normalizing the attributes and weighting them it is possible to obtain an index, even from subjective attributes (Butler, 2002). In Figure 4.3, a formula based on the threat index proposed in the SAEM analysis is shown. This formula calculates the security index of an appliance for one protocol.

$$S.I.(Appliance, Protocol) = M \times \big(V \times (1 - P)\big) \times F$$

S.I: Security Index of an appliance for a protocol

M: Number of packets of the protocol

V: Capacity of the security appliance to analyse the protocol (0 or 1)

P: Percentage of successful attacks from a particular protocol that the security appliance did not prevent (from 0 to 1)

F: Weight calculation for the protocol (from 0 to 1)

Figure 4.3: Security Index formula

By applying this formula to each appliance, the values for each protocol can be added and finally, it will give a total score. The final step is shown in Table 4.8.

Table 4.8: Calculation of the security index

	Protocol 1	**Protocol 2**	**TOTAL**
Appliance 1	S.I. (A1, P1)	S.I. (A1, P2)	J = S.I. (A1, P1) + S.I. (A1, P2)
Appliance 2	S.I. (A2, P1)	S.I. (A2, P2)	K = S.I. (A2, P1) + S.I. (A2, P2)
		Total security score	J + K

4.3.4 Analysis of security upgrades

The vulnerability and security scores of the network are a start point. They can be used as a reference to measure the improvement an upgrade can do to the scores. The vulnerability score will change if a protocol is replaced by another or not used anymore. If a new protocol is added, it would not count into the scoring because there is no information about how many packets of that protocol will travel through the network.

The final result of the vulnerability score will consider the packets of the replaced protocol to be the ones from the new protocol. Even if the new protocol uses more packets than the previous one, it is only done to maintain the relationship between the vulnerabilities and the presence of the previous protocol in the network. In Table 4.9, an example is shown where protocol 2 from Table 4.4 is replaced by a new protocol.

Table 4.9: Example of the new vulnerability score

	Protocol 1		Protocol 3		Previous Score	New Score
	Percentage of vulnerability score	Percentage of packets	Percentage of vulnerability score	Percentage of packets		
Low	$A1 = a1/E$	$L1 = \frac{M}{M+N}$	$A3 = a3/E$	$L2 = \frac{N}{M+N}$	$(\boldsymbol{A1} \times \boldsymbol{L1}) + (\boldsymbol{A2} \times \boldsymbol{L2})$	$(\boldsymbol{A1} \times \boldsymbol{L1}) + (\boldsymbol{A3} \times \boldsymbol{L2})$
Medium	$B1 = b1/E$		$B3 = b3/E$		$(\boldsymbol{B1} \times \boldsymbol{L1}) + (\boldsymbol{B2} \times \boldsymbol{L2})$	$(\boldsymbol{B1} \times \boldsymbol{L1}) + (\boldsymbol{B3} \times \boldsymbol{L2})$
High	$C1 = c1/E$		$C3 = c3/E$		$(\boldsymbol{C1} \times \boldsymbol{L1}) + (\boldsymbol{C2} \times \boldsymbol{L2})$	$(\boldsymbol{C1} \times \boldsymbol{L1}) + (\boldsymbol{C3} \times \boldsymbol{L2})$
Critical	$D1 = d1/E$		$D3 = d3/E$		$(\boldsymbol{D1} \times \boldsymbol{L1}) + (\boldsymbol{D2} \times \boldsymbol{L2})$	$(\boldsymbol{D1} \times \boldsymbol{L1}) + (\boldsymbol{D3} \times \boldsymbol{L2})$

On the other hand, the security score will be included in the Protocol protection matrix and the calculation of the security score. However, the Security Index formula will be different. Figure 4.4 shows the difference in the calculation formula. In the formula, there is no knowledge of previous attacks because the upgrade is not installed yet. Additionally, a new variable is defined to represent a condition: if there are already two security appliances protecting a protocol, the variable takes a value of zero. The reason is that a new upgrade for a well-protected protocol does not add much value in terms of defence.

$$S.I.(Upgrade, Protocol) = M \times (V \times T) \times F$$

S.I: Security Index of an upgrade for a protocol

M: Number of packets of the protocol

V: Capacity of the security appliance to analyse the protocol (0 or 1)

T: A value that confirms if there are less than two upgrades protecting the protocol (0 or 1)

F: Weight calculation for the protocol (from 0 to 1)

Figure 4.4: Security Index formula for an upgrade

4.3.5 Analysis of the procedure to understand the scores

The model was able to provide with two important outputs:

a) The vulnerability scores
b) The security index

Both numbers have an input metric in common: the number of packets for each protocol. However, because they suffered different processes, they cannot be treated together without losing their meaning.

From both outputs, the security index is easier to analyse, as it produces one unique index similar to the SAEM analysis. Furthermore, the vulnerability score does not describe an improvement in protecting protocols. It only describes changes in the protocols and vulnerabilities inside the network. Furthermore, the security index considers the relevance of some protocols for the IT managers.

With the previous analysis, the security index could be stated as the first output metric to consider when deciding on an upgrade. In case there is a similar security index value between two upgrades, the vulnerability scores can be used to differentiate them.

5 Experimental plan

Following the outcomes of the previous analysis, to fully evaluate the proposed model, it is imperative to test it accordingly and analyse its behaviour in assessing security upgrades. In order to successfully test the model, it is necessary to apply it in a scenario with the necessary metrics inside to asses it. For that purpose, information will be gathered from inside a TCP/IP network.

The next part of the research question looks into assessing how beneficial a security upgrade can be for a network. The assumption that the upgrades have not been deployed will make them exempt from emulating.

From the simulation, the protocols inside it will be identified. After obtaining the vulnerability and security index scores from the network, some possible upgrades defined by the findings of previous experiments will be introduced into the model. Finally, the model will state the new scores for the system for each upgrade, so it can be compared with the initial metric and therefore establishes the most beneficial upgrade.

5.1 Generating the network environment

The goal of the model is to assess the benefits of different security upgrades and compare them. For that purpose, a network should be the foundation for any proposed upgrade. As mentioned in Section 1.1 and 1.2, the model is oriented to a TCP/IP network and more precisely to communication protocols. The model is not concerned about the operative system and applications used inside the clients. The main purpose of the network is to be able to produce or replicate communication protocols. A network able to transmit these protocols could be made by using real network elements or by using a virtual environment or simulation.

In Section 2.1.3 from the Research Methods, it was discussed the pros and cons of using a virtual environment or a real one. From the analysis, the use of a virtual environment was preferred as it provides more control and requires fewer resources than acquiring the

elements to build a real network.

A virtual network should take into consideration the topology and platforms running in real networks. It should also allow techniques for network scanning including host discovery, port states scanning and operating system identification (Zhai & Wang, 2011). Port scanning and identifying the operative system are not the main concerns in the experiment but it could prove beneficial for other researchers in making their own experiments.

Durkota, et al. (2016) used a case study from the Swedish Defence Research Agency, the elements considered to test their model were a router, a mail and web server, firewalls and users. The purpose of this case study was to test their proposed model. A topology of the case study is shown in Figure 5.1.

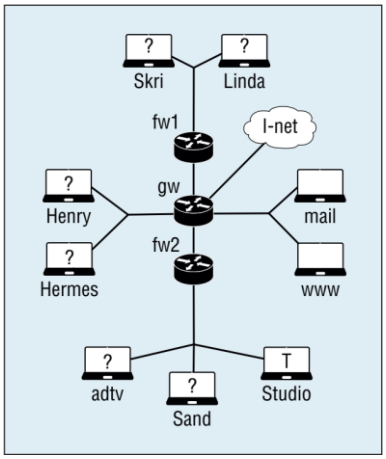


Figure 5.1: TV company's network used in a case study (Durkota, et al., 2016)

Ashtiani & Abdollahi Azgomi (2014) indicate different case studies to test their Distributed Cyber Attack Simulator (DCAS). While they were created to test particular attacks, they also present a larger network for complex attacks. In their research, they introduce new elements to consider: An Intrusion Prevention System (IPS) and a Domain Name System (DNS) server. The IPS is used to match a signature's database with the packets being sent to a target while the DNS is used for a DNS spoofing attack scenario. The DNS server should be included for internal hosts to resolve domain names. The IPS could be considered as a security upgrade and it will be reviewed in the following sections.

Sawilla & Burrell (2010) elaborates two case studies for testing their algorithms used for improving security in networks. In their first case study (Figure 5.2), they provide a topology for a multi-step attack where an attacker tries to get access to a File Server by reaching a Web and Mail server first:

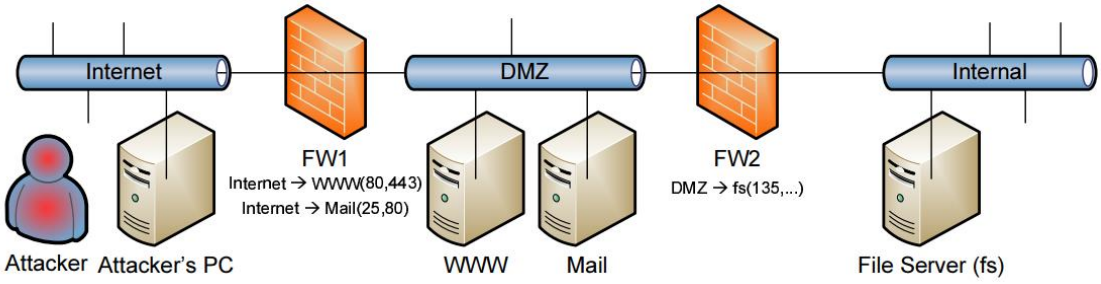


Figure 5.2: Example of a multistep attack (Sawilla & Burrell, 2010)

While Durkota, et al. (2016) also considers more than one firewall and the use of a Web and Mail server, this new case study creates intermediate elements to safeguard an important element in the network. At the same time, it shows a File Server as an important asset for an organization. A File Server shares documents between employees, making it crucial for daily processes.

Their second case study considers a network based on a real one. Similar to the first case study, the attacker is outside the network. In Figure 5.3, the topology is presented:

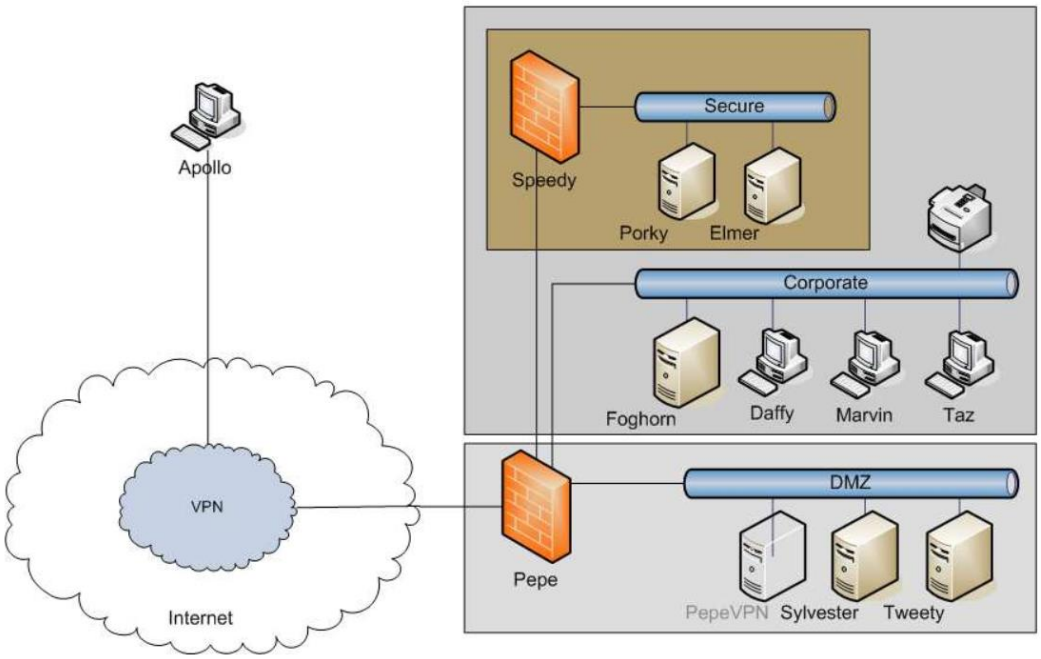


Figure 5.3: Topology based on a real network (Sawilla & Burrell, 2010)

Pepe (perimeter firewall) hosts a VPN service for remote users. The purpose of the other servers is explained in Table 5.1.

Table 5.1: Description of servers and their purpose inside the network (Sawilla & Burrell, 2010)

Server's Name	Applications	Purpose
Porky Server	Citrix Server	• Remote application hosting • Hosting the company's financial info
Elmer Server	File and Database Server	• File and source repository • SQL database with sensitive data
Foghorn Server	Mail and Web Application Server	• Internal mail • Web applications
Sylvester Server Tweety Server	HTTP, DNS and Mail Server	• Host web pages • Resolve domain names • Sendmail service

From Elmer, the main purpose for them is to hold files and data. The Remote application server (Porky Server) from Citrix, also known as XenApp, allow a series of connections

for clients, administrators and other services (Citrix Systems, Inc. , 2018). It is not specified how the financial information is hosted, however, it is an indicator of how important this server is. While only the communication ports are mentioned by CITRIX (2018), some of the services are stated or implied: SQL, HTTP, HTTPS, Wake-On-LAN, LDAP, etc. However, only TCP ports 80 and 443 are used for clients to access the remote Apps (Citrix Systems, Inc., 2015). Since the communication protocols are already present in the web server, adding this server into the experiment will be redundant for the purpose of the research.

The terminals inside the network run several operative systems. The reason behind that approach could be to represent exploitable vulnerabilities inside the software. However, the goal of the attacker is to gain access to the servers. Finally, the VPN service can be seen as another service and not as an upgrade. If a network does not have a connection for remote users, then adding the VPN service is not improving but assuring the security.

Each network provides with different elements from testing models and different attack simulations. However, by generating a new environment based on some of the elements mentioned before and taking into consideration the topology of the first experiment in Sawilla & Burrell (2010), a less complex network topology can be proposed.

In Figure 5.4, the proposed topology is presented. It considers the multipath attack by using two firewalls to defend the key element in the network: the users. Two workstations represent the users, and two servers are added to add services to the network. Although other elements like a Database server or a File Server could be added to include protocols like SQL or FTP travelling through the network., a complex network requires more effort to build and takes focus and time from testing the model. Furthermore, during the research, the tool used to emulate the network stopped having support. As a result, it was not possible to install other services.

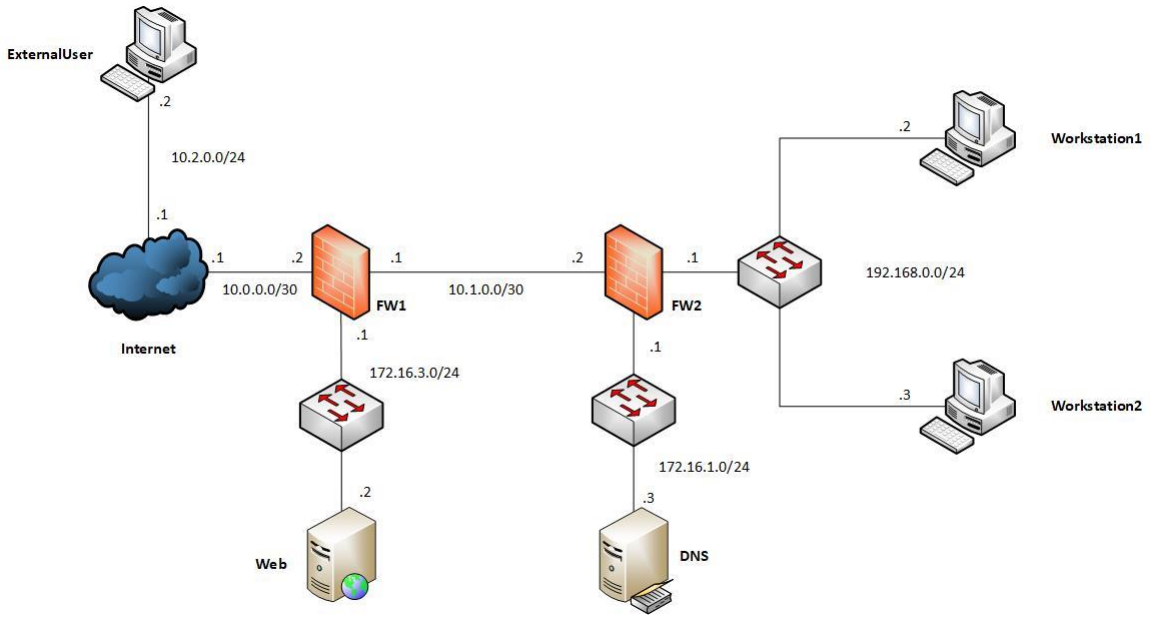


Figure 5.4: Network topology of the experiment

5.1.1 Tools used in the experiment

During this research, three options were available for the generation of the virtual network: Netkit, GNS3 and EVE. More details about these tools are mentioned in Appendix G. From the three, Netkit does not need any licensed operative system or high system requirements. Furthermore, as the experiment is more focused on the terminals and servers and the traffic between them, it is not needed to emulate a router or element from a particular manufacturer.

The version of the tool is *Netkit-NG 3.0.4*. As mentioned in the resources page, the relevant characteristic for using this tool is that the filesystem and kernel can be updated and upgraded, and new packages can be installed for installing more services inside it (Iguchi-Cartigny, 2014). However, since May 31 of 2018, the operative system behind each terminal stopped being supported (Debian, 2018) which makes changing the filesystem a big challenge. Because this change occurred during the experiment, the provisioning of the machines considered only the services already provided by *Netkit-NG* and any service that could not require too much complexity to deploy.

To able the experiment to be reproduced and add portability, and because the tool requires a Linux environment, *Ubuntu 18.04 LTS* will be installed on a virtual machine

inside *Oracle VM Virtualbox 5.2.12*. Both tools (*Ubuntu* and *Oracle VM Virtualbox*) can be changed as the experiment do not rely exclusively on those tools. Other important tools to consider are:

- *Wireshark 2.6.1* for analysing the capture packets between the communication of the elements in the network
- *Nmap* for host detection and port status. This application is already provided by Netkit-NG and working in each terminal, so a port scan can be performed from any element of the network

5.1.2 Generation of the experimental environment

The environment considers the topology stated in Section 5.1 for a hypothetical company. The company runs different services inside the domain "company.test". After implementing the Virtual Machine and deploying *Netkit*, an application for each server will provide the communication protocols needed for testing the model.

As a first step, *Ubuntu 18.04 LTS* was installed on a virtual machine inside *Oracle VM Virtualbox 5.2.12* where the *Netkit-NG* software was deployed. The specifications of the virtual machine are:

- Architecture: 64 bits
- Base memory: 11264 MB
- Number of Processors: 4 CPUs
- Storage: 10 GB
- System name: computer

Netkit-NG 3.0.4 is considered the latest stable release at this point in time (Iguchi-Cartigny, 2014). For its installation it comprehends 3 files:

- Core version: netkit-ng-core-32-3.0.4.tar.bz2
- Filesystem: netkit-ng-filesystem-i386-F7.0-0.1.3.tar.bz2
- Kernel: netkit-ng-kernel-i386-K3.2-0.1.3.tar.bz2

After downloading the packages, the files should be decompressed in the Home directory (`/home/computer`) of the system. Table 5.2 presents the terminal commands needed to install Netkit.

Table 5.2: Installation steps for Netkit (NETKIT, 2016)

STEP	COMMAND	PURPOSE / OUTCOME

1	After opening a terminal:
	`tar -xjSf netkit-ng-core-32-3.0.4.tar.bz2`
	`tar -xjSf netkit-ng-filesystem-i386-F7.0-0.1.3.tar.bz2`
	`tar -xjSf netkit-ng-kernel-i386-K3.2-0.1.3.tar.bz2`
		The files are decompressed. All the files should be in the same directory (for this case /home/computer)
		A new directory will be created: netkit

2	`gedit ~/.bashrc`	This command opens the bash file

3	`export NETKIT_HOME=/home/computer/netkit-ng`
	`export MANPATH=:$NETKIT_HOME/man`
	`export PATH=$NETKIT_HOME/bin:$PATH`
		The variable NETKIT_HOME is configured according to the place Netkit has been decompressed.
		
		Add the following at the end of the file (delete any space in the path):

4	`. $NETKIT_HOME/bin/netkit_bash_completion`	(Optional) Allows to autocomplete Netkit commands
5	`source ~/.bashrc`	This command resets the bash file
6	`sudo apt-get install xterm wireshark make net-tools`	Provides the default terminal and important tools for Netkit to initialize. It is needed to choose YES for any option presented during the installation
7	`sudo apt-get install libc6-i386 lib32ncurses5 lib32z1`	These libraries help to run 32bit programs

To check the correct installing, the following script is founded in the distribution:

```
cd /home/computer/netkit
./check_configuration.sh
```

After selecting the elements to be considered on the network in Section 5.1, a file was created in the home directory: /home/computer/experiment and after going inside the directory, the following commands are executed to create the files inside the folder:

```
mkdir DNS ExternalUser FW1 FW2 Internet Web Workstation1 Workstation2
touch DNS.startup ExternalUser.startup FW1.startup FW2.startup
Internet.startup Web.startup Workstation1.startup Workstation2.startup
touch lab.conf lab.dep
```

Figure 5.5: Configuration commands to create the initial files for the experiment

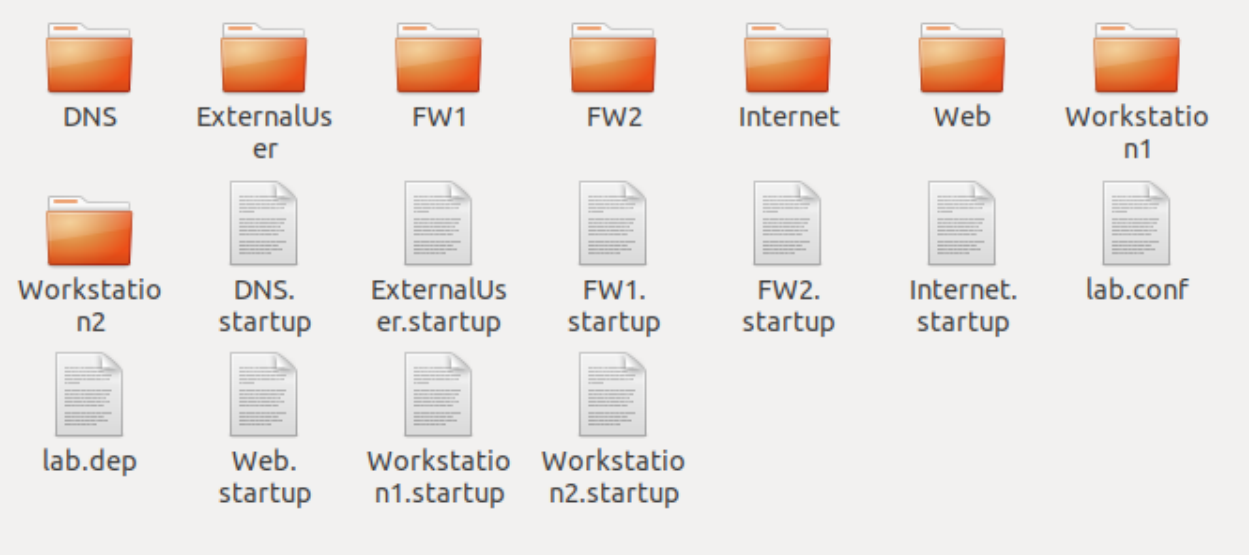


Figure 5.6: Example of the Directory File structure of the experiment

The file *lab.conf* contains the detail of the connections between the elements, while the files with *.startup* extension define the initial commands to be deployed on a particular network element when it boots up. Each File Directory contains all the data needed for the services inside each element to run properly and according to the experiment. The final configuration files, including the ones mentioned later in this experiment, can be seen in Appendix H.

Netkit-NG has different services pre-installed, so they can be used in the experiment. By reusing some of the files inside the Netkit-NG filesystem, time can be saved. Some other services do not require their files to be modified.

Table 5.3: Configuration example of DNS Server

STEP	COMMAND	PURPOSE / OUTCOME
1	`cd /home/computer/experiment` `lstart DNS`	By standing in /home/computer/netkit-ng the command starts the DNS Server.
2	In DNS Xterm terminal `cp /etc/dnsmasq.conf /hosthome`	It copies the file dnsmasq.conf in the filesystem to /home/computer/
3	lcrash DNS	The DNS server is stopped
4	Modification of the dnsmasq.conf Creation of dnsmasq_hosts.conf	The configuration files used for the DNS service are modified (according to Appendix H). Additionally, a new file is created to simplify the configuration process.
5	`mkdir DNS/etc`	Creates a folder in the DNS directory. Any file copied to "etc" will overwrite the files in the common filesystem used by Netkit-NG for that terminal only.
6	`mv /home/computer/dnsmasq.conf DNS/etc` `mv /home/computer/dnsmasq_hosts.conf DNS/etc`	After doing any change on the file, it can be moved to the corresponding directory for the next startup

7	On the DNS.startup file add the following in the end
	`#Start the services`
	`service dnsmasq start`
		In order to start the service when the lab is starting, the service is declared in the startup file.

8	`cp /etc/resolv.conf resolv.conf`
	`gedit resolv.conf`
	Inside the file, the only uncommented line should be:
	`nameserver 172.16.1.3`
		The file in the host system is used as a reference. From the copy on the home directory, the nameserver is modified to point the IP of the DNS server (172.16.1.3)

Table 5.3 (Continued): Configuration example of DNS Server

9	`mkdir <element>/etc`
	`cp resolv.conf <element>/etc`
	`rm resolv.conf`
		The file should be copied in each element inside the company's network (all except ExternalUser and Internet). Repeat this step for each element.

10	`touch hosts`	A text file named "hosts" is created.

11	Add the following in "hosts":
	`127.0.0.1 localhost`
	`127.0.1.1 <element>.company.test <element>`
		For each element inside the network, the file is modified to indicate domain where the element resides

12	`cp hosts <element>/etc`	Steps 11 and 12 should be done for each element inside the network
´13	`rm hosts`	Eliminates the template

In Appendix 2, the contents of dnsmasq.conf and dnsmasq_hosts.conf are presented. After starting the laboratory with the "lstart" command, the configuration can be validated by using the commands shown in Figure 5.7:

```
root@Workstation2:~# ping -c4 web.company.test
PING web.company.test (172.16.3.2) 56(84) bytes of data.
64 bytes from web.company.test (172.16.3.2): icmp_req=1 ttl=62 time=0.572 ms
64 bytes from web.company.test (172.16.3.2): icmp_req=2 ttl=62 time=0.996 ms
64 bytes from web.company.test (172.16.3.2): icmp_req=3 ttl=62 time=0.791 ms
64 bytes from web.company.test (172.16.3.2): icmp_req=4 ttl=62 time=0.878 ms

--- web.company.test ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3017ms
rtt min/avg/max/mdev = 0.572/0.809/0.996/0.156 ms
root@Workstation2:~#
```

Figure 5.7: Results of validating the DNS configuration

For the web server, the application is already installed in the Netkit-NG filesystem. Table 5.4 shows the commands needed to start the service:

Table 5.4: Configuration of the Web server

STEP	COMMAND	PURPOSE / OUTCOME

1	In Web.startup add the following:
	`#Start the services`
	`service apache2 start`
		By adding the following in the startup file, the web service will start when the lab runs

2	In Workstation1 Xterm terminal `curl http://web.company.test`	This command will validate the connection. The result is visible in Figure 5.8

```
Workstation2 login: root (automatic login)
Linux rootstrap 3.2.54-netkit-ng-K3.2 #2 Tue Nov 11 11:42:04 CET 2014 i686
Welcome to Netkit

root@Workstation2:~# curl http://web.company.test
<html><body><h1>It works!</h1>
<p>This is the default web page for this server.</p>
<p>The web server software is running but no content has been added, yet.</p>
</body></html>
```

Figure 5.8: Validation of the web service

5.2 Definition of security solutions for the network environment

The network generated has two security appliances installed as a foundation. FW1 in Figure 5.4 is used as a security perimeter to stop outside attacks and uses Network Address Translation (NAT) for external users to connect to the internal Web server. FW2 adds a second line of defence between external users and the DNS server. It also controls users from freely accessing both servers.

From Ashtiani & Abdollahi Azgomi (2014) and Sawilla & Burrell (2010), two solutions are proposed: an Intrusion Prevention System (IPS) and a Virtual Private Network (VPN) server for remote connection. An IPS activated in defensive mode with a database of attacks is capable of applying countermeasures to known threats (Ashtiani & Abdollahi Azgomi, 2014).

A VPN builds over an IP network to protect the communication between two peers over an insecure network. However, because there is no service in charge of connecting remote

users, a VPN solution cannot be considered an upgrade but a new service.

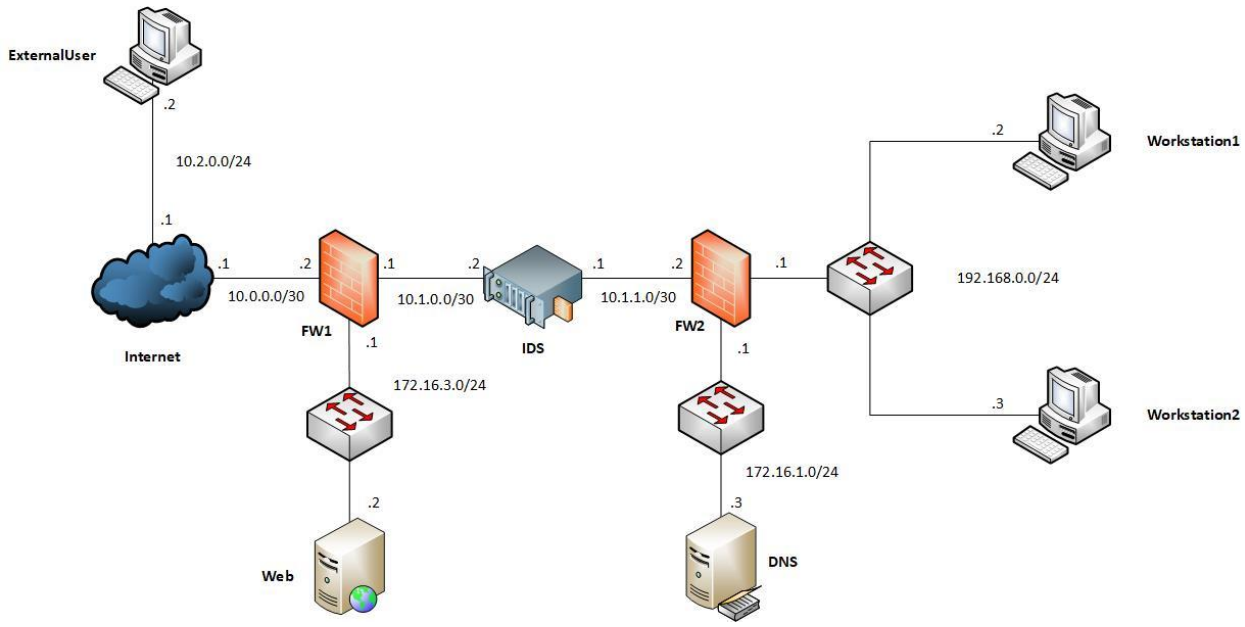


Figure 5.9: Security upgrade proposal – Intrusion Prevention System

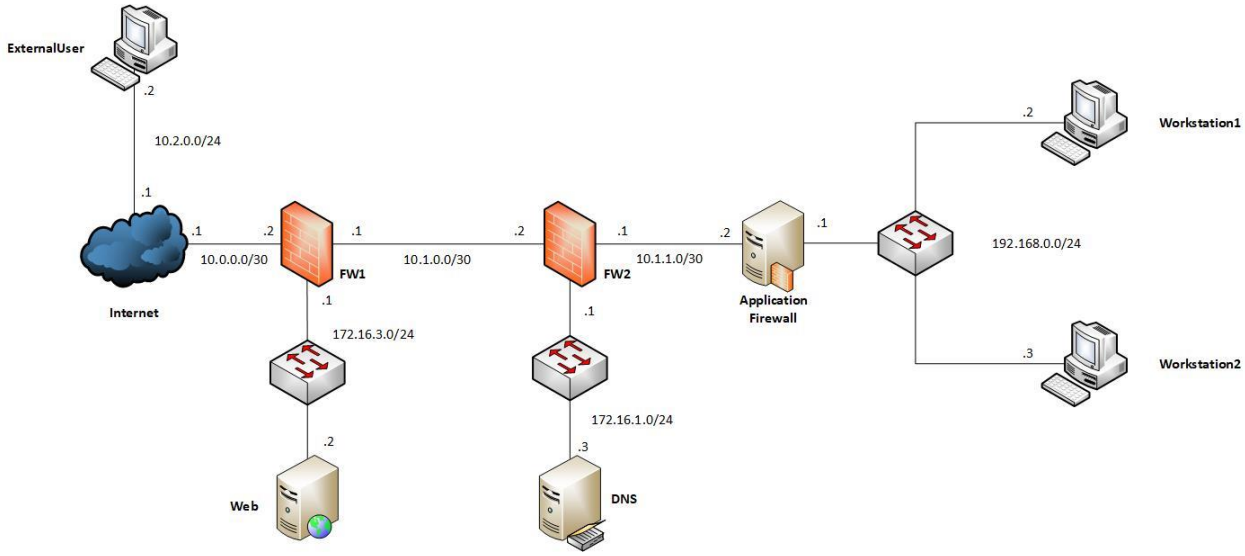


Figure 5.10: Security upgrade proposal – Application Firewall

Finally, other possible upgrades can be considered like HTTPS, firewalls working on an application level, and others related to network protection. (Canavan, 2001), including the filtering of one of the protocols found in the experiment.

5.3 Application of the model

For applying the model, first, a monitoring point is established. By considering the topology of the experiment, the assets to protect in this case are the users so the monitoring point should be as close as possible to those assets. Figure 5.11 shows where the monitoring will take place.

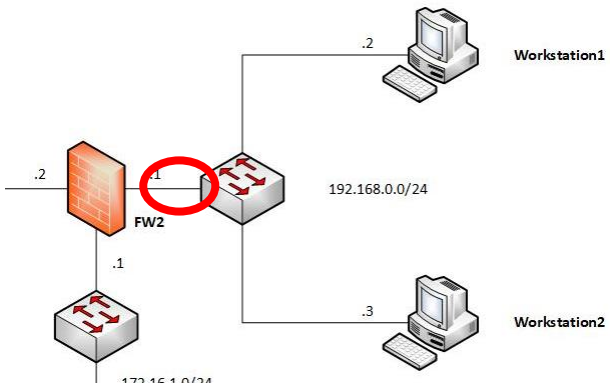


Figure 5.11: Monitoring point

Afterwards, because there are no records in the experimental network, it is necessary to produce traffic. Random traffic should be generated by the users in the network for analysis. In Table 5.5, the steps to generate traffic and the result are presented:

Table 5.5: Procedure to generate random traffic

STEP	COMMAND	PURPOSE / OUTCOME

1	In a new terminal (home directory of the user):
	`cd /home/computer`
	`touch ping.sh http.sh randomtraffic.sh`
		It will create the scripts with commands to generate traffic

2	`chmod 777 ping.sh randomtraffic.sh http.sh`	This command will enable the scripts to be executable
3	`echo "curl http://web.company.test" > http.sh` `echo "ping -c1 10.0.0.1" > ping.sh`	New commands are added to ping.sh and http.sh

Table 5.5 (Continued): Procedure to generate random traffic

4	Open randomtraffic.sh and add the following ten times (Down, 2018): `if ((RANDOM % 2)); then /hosthome/ping.sh; else /hosthome/http.sh; fi`	This command will randomly execute http.sh or ping.sh ten times.
5	`cd experiment` `lstart`	The next step is starting the lab
6	On FW2: `tcpdump -i eth0 -w /hosthome/monitoring.pcap`	FW2 starts monitoring the interface that points to the users
7	On Workstation1 and Workstation2 `/hosthome/randomtraffic.sh`	Both workstations create random traffic for analysis

The result of the process is a file with the traffic capture by FW2 in the folder directory. By opening the file and going to `Statistics>Protocol Hierarchy`, the number of packets for each protocol is visible.

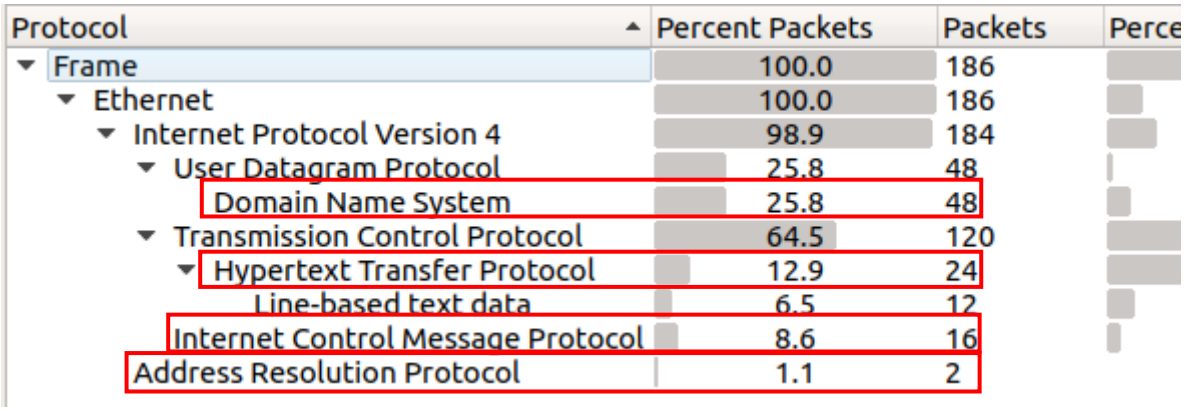


Figure 5.12: Extract of packets

Table 5.6: Number of packets for each protocol

	HTTP	DNS	ARP	ICMP
Number of packets	24	48	2	16

5.3.1 Calculation of the Vulnerability Score

The next step is to obtain information from the Vulnerability Database. In this experiment, the National Vulnerability Database is used. It provides information on vulnerabilities and scores according to the CVSS.

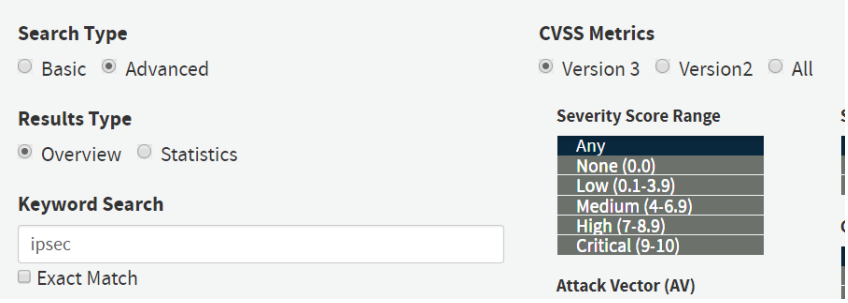


Figure 5.13: Extract of the National Vulnerability Database search webpage (National Institute of Standards and Technology, 2018)

The results of the vulnerabilities of each protocol according to the steps shown in Table 4.2 are calculated in Tables 5.7 to 5.10

Table 5.7: Amount of vulnerabilities in HTTP

HTTP			
Range	**Number of vulnerabilities**	**Amount of vulnerabilities in each range**	**Percentage**
Low	526	0.01785108	2%
Medium	11122	0.37745198	38%
High	13177	0.44719338	45%
Critical	4641	0.15750356	16%
TOTAL	**29466**	**1**	**100%**

Table 5.8: Amount of vulnerabilities in DNS

DNS			
Range	**Number of vulnerabilities**	**Amount of vulnerabilities in each range**	**Percentage**
Low	4	0.02366864	2%
Medium	47	0.27810651	28%
High	89	0.52662722	53%
Critical	29	0.17159763	17%
TOTAL	**169**	**1**	**100%**

Table 5.9: Amount of vulnerabilities in ARP

ARP			
Range	**Number of vulnerabilities**	**Amount of vulnerabilities in each range**	**Percentage**
Low	0	0	0%
Medium	10	0.25	25%
High	21	0.525	53%
Critical	9	0.225	23%
TOTAL	**40**	**1**	**100%**

Table 5.10: Amount of vulnerabilities in ICMP

ICMP			
Range	**Number of vulnerabilities**	**Amount of vulnerabilities in each range**	**Percentage**
Low	1	0.03571429	4%
Medium	5	0.17857143	18%
High	16	0.57142857	57%
Critical	6	0.21428571	21%
TOTAL	**28**	**1**	**100%**

Finally, the vulnerability score according to the process described in Section 4.3.2 is presented in Table 5.11.

Table 5.11: Vulnerability score for each of vulnerabilities

	HTTP		DNS		ARP		ICMP		Total AVG
	Vuln Score	% of packets	Vuln Score	% of packets	Vuln Score	% of packets	Vuln Score	% of packets	
Low	0.02	0.267	0.02	0.533	0.00	0.022	0.04	0.178	**0.02**
Medium	0.38		0.28		0.25		0.18		**0.29**
High	0.45		0.53		0.53		0.57		**0.51**
Critical	0.16		0.17		0.23		0.21		**0.18**

5.3.2 Calculation of the Security Index

Initially, the ranking of the protocols is done according to the model. After ranking the protocols according to the values presented by the author, the weight of each protocol can be seen in Table 5.12

Table 5.12: Weights for each protocol in the experiment

	HTTP	DNS	ARP	ICMP
Rank [1 - 100]	100	80	20	10
Weight	0.48	0.38	0.10	0.05

The next step is checking the records of previous attacks. Because in this network there are not attack logs, it can be assumed that the firewalls are protecting the protocol flawlessly. With that in mind, the protocol protection matrix is delivered.

Table 5.13: Protocol protection matrix

	HTTP	DNS	ARP	ICMP
FW1	0	0	0	1
FW2	0	0	1	1

The table shows that FW1 and FW2 are capable to protect ARP and ICMP, but ARP packets from and to the users are only present in FW2. However, even if they are able to filter ports, both firewalls are not able to analyse application protocols.

Finally, the values are analysed to produce the security index score.

Table 5.14: Security Index

	FW1	FW2	Total
HTTP	0	0	0
DNS	0	0	0
ARP	0	0.19047619	0.19047619
ICMP	0.76190476	0.76190476	1.52380952
		TOTAL	**1.71428571**

5.3.3 First upgrade: Implementation of HTTPS

As a first upgrade, the HTTP protocol will be replaced by HTTPS as the first approach of a security upgrade. In this case, the security upgrade is replacing a protocol, so the vulnerabilities of HTTP are replaced by the vulnerabilities from HTTPS.

Table 5.15: Amount of vulnerabilities in HTTPS

HTTPS			
Range	**Number of vulnerabilities**	**Amount of vulnerabilities in each range**	**Percentage**
Low	328	0.01346083	1%
Medium	8904	0.36541224	37%
High	10985	0.45081463	45%
Critical	4150	0.17031231	17%
TOTAL	**24367**	**1**	**100%**

Table 5.16: Change in the Vulnerability score by replacing HTTP with HTTPS

	HTTPS		DNS		ARP		ICMP		Total AVG	Change
	Vuln Score	Number of packets	Vuln Score	Number of packets	Vuln Score	Number of packets	Vuln Score	Number of packets		
Low	0.01	0.267	0.02	0.533	0.00	0.022	0.04	0.178	0.02	-5%
Medium	0.37		0.28		0.25		0.18		0.28	-1%
High	0.45		0.53		0.53		0.57		0.51	0%
Critical	0.17		0.17		0.23		0.21		0.18	2%

The last column shows how much the score has changed by replacing protocols. There is little change in general to the scores per range. This result does not mean that using

HTTPS is not important. It shows how much change the protocol can make into the network and compare it with other solutions.

5.3.4 Second upgrade: Filtering ICMP

Another option is discarding all the ICMP packets and taking out the protocol. It takes out the vulnerabilities involved in that protocol and the formula is recalculated using the remaining protocols:

Table 5.17: Change in the Vulnerability score by filtering ICMP

	HTTPS		DNS		ARP		Total AVG	Change
	Vuln Score	Number of packets	Vuln Score	Number of packets	Vuln Score	Number of packets		
Low	0.01	0.324	0.02	0.648	0.00	0.027	0.02	-11%
Medium	0.37		0.28		0.25		0.31	8%
High	0.45		0.53		0.53		0.50	-2%
Critical	0.17		0.17		0.23		0.17	-5%

5.3.5 Third upgrade: Intrusion Prevention System

An IPS is a security appliance that changes the security index. It does not replace or filters a protocol but protects a set of protocols. The first step is modifying the protocol protection matrix by adding the IPS.

Table 5.18: Protocol protection matrix by using IPS

	HTTP	DNS	ARP	ICMP
FW1	0	0	0	1
FW2	0	0	1	1
IPS	1	0	0	0

Because of the place where the IPS is installed, only HTTP is protected by the IPS. The new security index is shown in Table 5.19.

Table 5.19: New security index by applying the IPS

	FW1	FW2	IPS	Total
HTTP	0	0	11.4285714	11.4285714
DNS	0	0	0	0
ARP	0	0.19047619	0	0.19047619
ICMP	0.76190476	0.76190476	0	1.52380952
			TOTAL	**13.3333333**
			CHANGE	**600%**

5.3.6 Fourth upgrade: Application Firewall

An application firewall has the capacity to analyse application protocols as well as the IPS. If installed in the same place as the IPS, it will have the same effect: It will not be able to protect the DNS queries from the users going to the DNS server. However, by installing the application firewall near the users, the protocol protection matrix and the security index has new values. Furthermore, the application firewall can protect all the protocols seen in the Monitoring point.

Table 5.20: Protocol protection matrix by using an application firewall

	HTTP	DNS	ARP	ICMP
FW1	0	0	1	1
FW2	0	0	1	1
App FW	1	1	1	1

Table 5.21: New security Index by applying the application firewall

	FW1	FW2	App FW	Total
HTTP	0	0	11.4285714	11.4285714
DNS	0	0	18.2857143	18.2857143
ARP	0	0.19047619	0.19047619	0.38095238
ICMP	0.76190476	0.76190476	0	1.52380952
			TOTAL	**31.6190476**
			CHANGE	**1744%**

The application firewall can protect ICMP too but because there are already two appliances guarding the protocol, the application firewall does not add additional value to that protocol.

6 Analysis and Discussion

In the previous chapter, the experiment allowed obtaining scores from different upgrades. In this chapter the results are going to be compared, discussing how the score values allow visualizing the state of the network.

In the last part of the chapter, other models will be used. The score results of those models will be discussed and compared with the previous conclusions made with the proposed model.

6.1 Analysis of the results

From the four different upgrades, two are related to changes in the vulnerability score and the other two to the security index. As mentioned during the model proposal, the security index has more weight than the vulnerability score so the upgrades that change the security index will be evaluated first as the most appealing options.

Table 6.1: Comparison of the security index scores

Upgrade	Security Index Score	Percentage of change in the value
IPS	13.143	667%
Application Firewall	31.619	1744%

By looking at Table 5.21, the application firewall seems a better option than the IPS. There are two possible reasons for these high values: currently, there is no protection for the protocols considered more important, and the position of the appliances makes a difference.

Regarding the position, if the application firewall was positioned between FW1 and FW2, it will be unable to give protection to the ARP and DNS packets sent by or to the users. This would have an effect on the percentage of change, which is the result that shows the improvement done in the network.

Table 6.2: Comparison of the vulnerability scores

Ranges	Score of replacing HTTP with HTTPS	Percentage of change when replacing HTTP with HTTPS	Score of filtering ICMP	Percentage of change when filtering ICMP
Low	0.02	-5%	0.02	-11%
Medium	0.28	-1%	0.31	8%
High	0.51	0%	0.50	-2%
Critical	0.18	2%	0.17	-5%

From the statistics, filtering ICMP decreases the high and critical overall score values. Even if there are more HTTP packets, filtering ICMP has more impact in the overall scores. From all the ranges, the medium overall score for filtering ICMP is highly increased meaning that without ICMP there is a higher quantity of packets related to vulnerabilities in the medium range of threat. This does not mean a bad outcome since the initial number of packets has changed so the vulnerabilities of other protocols become more prominent. In general, the score values for filtering ICMP indicates a decrease in the possibility of harmful vulnerabilities so it could be considered as a better option.

6.2 Comparison with other models

For applying other models into the experimental network, some assumptions are needed. As seen in the literature review, models can focus on the adversary or on the beliefs of the people in charge of the network. For that purpose, any missing data from the experiment will be stated during the analysis and different assumptions will be used so the assessment can be done.

6.2.1 Results of the CAESARS Framework model

This model does not indicate the tools to use or the mathematical approach to scoring the risks inside the system. However, it gives the steps to do a risk assessment of the experiment.

For the sensors subsystem, the *NMAP* tool will be used to explore each of the elements

inside the network. After the analysis, only one vulnerability was found.

```
root@FW1:~# nmap --script vuln Web.company.test

Starting Nmap 6.00 ( http://nmap.org ) at 2018-08-18 18:48 CEST
Nmap scan report for Web.company.test (172.16.3.2)
Host is up (0.000079s latency).
rDNS record for 172.16.3.2: web.company.test
Not shown: 999 closed ports
PORT   STATE SERVICE
80/tcp open  http
| http-vuln-cve2011-3192:
|   VULNERABLE:
|   Apache byterange filter DoS
|     State: VULNERABLE
|     IDs:  CVE:CVE-2011-3192  OSVDB:74721
|     Description:
|       The Apache web server is vulnerable to a denial of service attack when n
umerous
|       overlapping byte ranges are requested.
|     Disclosure date: 2011-08-19
|     References:
|       http://seclists.org/fulldisclosure/2011/Aug/175
|       http://nessus.org/plugins/index.php?view=single&id=55976
|       http://osvdb.org/74721
|_      http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-3192
MAC Address: 3A:B8:3F:5D:00:B2 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 15.39 seconds
```

Figure 6.1: Result of the vulnerability analysis in Web server

By using the National Vulnerability Database as the database subsystem, the vulnerability CVE-2011-3192 is considered High by the CVSS v2, giving it a value of 7.8 (National Institute of Standards and Technology, 2011). This result can be considered part of the risk scoring subsystem.

In this scenario, filtering ICMP, adding an IPS, or adding an application firewall would not change the score, as the vulnerability will continue being part of the Web server. However, changing HTTP for HTTPS can alter the scores. For that purpose, the commands in Figure 6.2 enable the Web server to start using HTTPS.

```
a2enmod ssl
a2ensite default-ssl
service apache2 restart


iptables -A INPUT -i eth0 -p tcp --dport 80 -j DROP
```

Figure 6.2: Enabling HTTPS instead of HTTP in the Web server (DigiCert, 2018)

The new outcome of the vulnerability scan shows that there are no more vulnerabilities in the system:

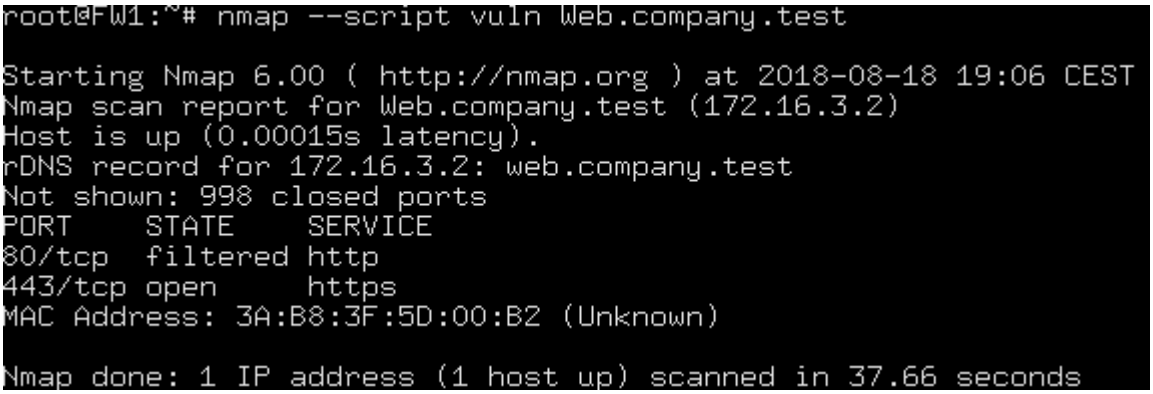


Figure 6.3: Vulnerability scanning of Web server after upgrading

6.2.2 Results of the SAEM analysis method

For this model, the first steps are oriented to obtain a threat index. However, these attributes are defined based on the criteria of the security and IT managers: outcome attributes, the frequency of those outcomes, weights for the outcomes, etc. As it is mentioned in the paper, their index is based on subjective estimates to quantify the experience of the security managers (Butler, 2002).

From this assessment, some of the threats mentioned can be assumed as part of the experiment. It is possible for a user to scan the network and look for vulnerabilities. Also, because there is no encryption or monitoring in the connection to the web server, there could be a possibility for risks and vulnerabilities related to browsing.

Finally, sensitive information in the communication can be analysed (including password nabbing). The results of the threat index including the identified risks are listed in Table 6.3.

Table 6.3: Threat indices for the experiment (Butler, 2002)

Threats	Threat Index
Scanning	886.44
Browsing	226.71
Password Nabbing	0.62
TOTAL	**1113.90**

With the treat indices result, the model can be applied. The model contemplates defence-in-depth which categorizes security upgrades in protection, detection and recovery purposes. Each possible technological upgrade can be in one or more categories.

The first step is assigning each upgrade into the categories. All the proposed security upgrades are focused on protecting the network protocols, however, the Intrusion Prevention System has also detection capabilities.

Table 6.4: Categories for the upgrades (Butler, 2002)

Protection	Detection	Recovery
Intrusion Prevention System Application Firewall Replacing HTTP on HTTPS Filtering ICMP	Intrusion Prevention System	

The second step is applying the technologies to cover the possible risks. From the possible upgrades, filtering ICMP does not cover any of the mentioned risks.

Table 6.5: Risks covered by the technologies

Risk	Security Technologies
Scanning	Application Firewall Intrusion Prevention System
Browsing	Replacing HTTP on HTTPS Intrusion Prevention System
Password Nabbing	Replacing HTTP on HTTPS

The last step is calculating the benefit estimates, which means estimating how effective each technology is covering each risk. In that case, the percentage determined how much the risk is reduced. To maintain consistency, some estimates are taken from the research study. However, for the IPS and the HTTPS upgrade, some of the estimates are not reflected in the study. For that purpose, the following assumptions are made:

- The IPS has the same capabilities as the Network IDS and Host IDS, as the browsing risk relates to vulnerabilities related to the communication.
- Because there is no information for HTTPS to relate to, it will be assumed that its effectiveness for browsing and password nabbing is 100%

Table 6.6: Effective estimates (Butler, 2002)

	Scanning	Browsing	Password Nabbing
Intrusion Prevention System	33%	50%	0%
Application Firewall	75%	0%	0%
Replacing HTTP on HTTPS	0%	100%	100%

Table 6.7: New Threat Indices

Threat	Intrusion Prevention System	Application Firewall	Replacing HTTP on HTTPS
Scanning	593.91	221.61	886.44
Browsing	113.36	226.71	0
Password Nabbing	0.62	0.62	0
Total	**707.89**	**448.94**	**886.44**
Change	**36%**	**60%**	**20%**

Another output of the SAEM analysis is the security coverage. By looking into the SAEM analysis study (Butler, 2002), Figure 6.4 can be constructed:

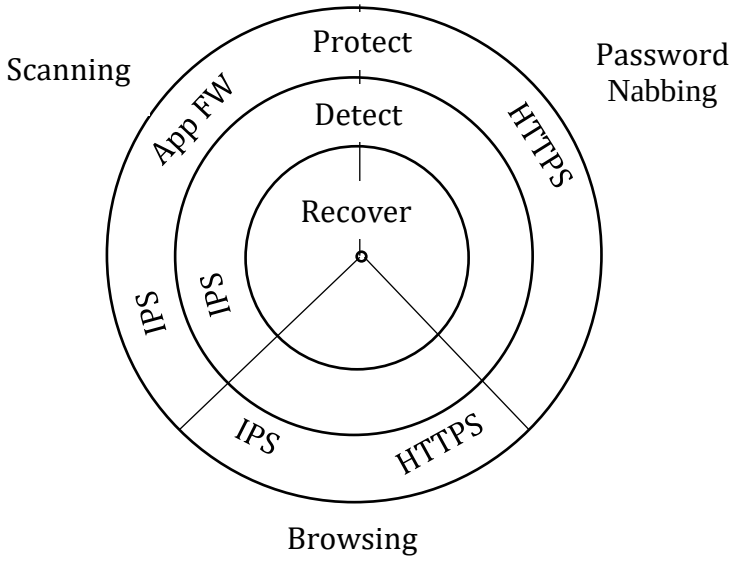


Figure 6.4: Security coverage of the upgrades

Other steps part of the analysis, like costs and sensitive analysis, were mentioned but not applied to their model.

Finally, it is not mentioned in the study which technology is the most suitable in the end, as the model is only intended to present the benefits of each technology to the security manager for a more informed decision.

6.2.3 Comparison of the results

From the results, the proposed model indicates that an application firewall is the best

option for protecting the communication protocols in the network, while the CAESARS framework indicates that HTTPS eliminates the vulnerabilities inside the network. However, the SAEM analysis does not provide a unique score or a comparison between the outputs provided.

In the SAEM analysis, the threat index has a better score by using an application firewall. However, in the security coverage, the IPS covers two dimensions. Because the focus has been the protection of the protocols, the threat index will be considered over the security coverage.

Table 6.8 compares the best solution proposed by each model and the reasons for that.

Table 6.8: Comparison of the results

Model used	Best option	Reason
Proposed model	Application Firewall	Has a better security index score than the others
CAESARS framework	Replacing HTTP with HTTPS	Eliminates the vulnerabilities inside the network (drops the score to zero)
SAEM analysis	Application Firewall	It lowers the threat index more than the others

When looking into CAESARS result, it can be seen that even if it is not oriented to the communication protocols travelling into the network, it assesses the vulnerabilities inside terminals or servers and proposes a much easier solution than the others. CAESARS framework does not find any problem in other services, while the proposed model considers every protocol as possible risks.

In that sense, the security index probes more valuable as CAESARS was not able to consider the security appliances inside the network. It could be possible that with establishing more elements of the CAESARS framework or changing the tools used in the experiment, the result would have been different.

Another critique to CAESARS framework is that for that approach to work, all the subsystems are needed to be implemented in the system, which could prove difficult depending on the network it is working.

SAEM analysis provided more than one score but, in the end, the security manager is the one in charge of prioritising the results and choosing which one is the most relevant. But also showed dimensions that the proposed model failed to analyse: detection and recovery.

A critique to SAEM analysis is the subjective values used through the process. A different opinion could have made another upgrade as the possible option. Furthermore, values like effectiveness depend greatly on the experience of the security manager. If in the future another person is in charge of assessing the security, the values can change drastically.

7 Conclusions and Future Work

The proposed model allows to assess the communication protocols inside a network and compare possible upgrades establishing the best solution. The model used a set of metrics and other models as a reference as seen in the literature review, and the results were compared to obtain the highlights and limitations of using it.

A virtual network, based on previous research and tests, was developed to fully analyse the proposed model as well as the other two models. While its purpose was for analysing the model, the network used in the experiment could be used by other researchers to test and analyse other possible models.

As one of the limitations, the model sees each protocol as vulnerable for the vulnerability score. However, the security index probes more valuable in assessing other factors rather than the vulnerabilities.

For future work, the model can be modified to allow vulnerability metrics related to software. By doing this, the model will assess communication protocols and/or software applications.

In terms of the virtual network, the kernel and filesystem of the tool used in the experiment can be updated. That would allow updating and installing new packages into the system and create new services inside. This can also allow installing particular versions of software with known vulnerabilities.

8 References

Arshed, N. & Danson, M., 2015. The Literature Review. In: K. O'Gorman & R. MacIntosh, eds. *Research Methods for Business & Management.* Oxford: Goodfellow Publishers Ltd, pp. 31-49.

Ashtiani, M. & Abdollahi Azgomi, M., 2014. A distributed simulation framework for modeling cyber attacks and the evaluation of security measures. *Simulation,* 90(9), pp. 1071-1102.

Barford, P. et al., 2010. Cyber SA: Situational Awareness for Cyber Defense. In: S. Jajodia, P. S. V. Liu & C. Wang, eds. *Cyber Situational Awareness - Issues and Research.* New York: Springer, pp. 3-13.

Bodeau, D., Fabius-Greene, J. & Graubart, R., 2010. *How Do You Assess Your Organization's Cyber Threat Level?,* USA: The MITRE Corporation.

Bombal, D., 2017. *Where do I get IOS images? - GNS3.* [Online] Available at: https://docs.gns3.com/1vJwh4_whwtfjb8pQ8vKekcWrA1galIqA1eHgeClOsPY/index.html
[Accessed 18 August 2018].

Butler, S. A., 2002. Security Attribute Evaluation Method: A Cost-Benefit Approach. *Proceedings of the 24th international conference on Software engineering,* pp. 232-240.

Canavan, J. E., 2001. *Fundamentals of Network Security.* Massachusetts: Artech House.

Citrix Systems, Inc. , 2018. *Communication Ports Used by Citrix Technologies.* [Online] Available at: https://support.citrix.com/article/CTX101810#XenDesktop XenApp
[Accessed 14 July 2018].

Citrix Systems, Inc., 2015. *Controlling Client Connections in XenApp.* [Online] Available at: https://docs.citrix.com/en-us/xenapp-and-xendesktop/xenapp-6/ps-sessions-wrapper-v2/ps-sessions-cfg-user-conn-all.html
[Accessed 14 July 2018].

Dainese, A., 2017. *Unified Networking Lab v2 (UNetLabv2) | Andrea Dainese.* [Online] Available at: http://www.routereflector.com/unetlab/
[Accessed 18 August 2018].

Debian, 2018. *LTS - Debian Wiki.* [Online] Available at: https://wiki.debian.org/LTS
[Accessed 1 August 2018].

DigiCert, 2018. *Ubuntu with Apache2: CSR & SSL Installation (Open SSL).* [Online]

Available at: https://www.digicert.com/csr-ssl-installation/ubuntu-server-with-apache2-openssl.htm
[Accessed 19 August 2018].

Down, C., 2018. *bash - Run commands at random - Unix & Linux Stack Exchange.* [Online] Available at: https://unix.stackexchange.com/questions/81566/run-commands-at-random
[Accessed 15 August 2018].

Duggan, D. P., Thomas, S. R., Veitch, C. K. K. & Woodard, L., 2007. Categorizing Threat: Building and Using a Generic Threat Matrix. *Sandia National Laboratories report SAND2007-5791.*

Durkota, K. et al., 2016. Case Studies of Network Defense with Attack Graph Games. *IEEE Intelligent Systems,* 31(5), pp. 24-30.

EVE-NG, 2017. *System Requirement.* [Online] Available at: http://www.eve-ng.net/documentation/installation/system-requirement [Accessed 18 August 2018].

Fielder, A. et al., 2016. Decision support approaches for cyber security investment. *Decision Support Systems,* Issue 86, pp. 13-23.

FIRST.org, 2018. *CVSS v3.0 Specification Document.* [Online] Available at: https://www.first.org/cvss/specification-document [Accessed 17 August 2018].

Galaxy Technologies, LLC, 2018. *Software / GNS3.* [Online] Available at: https://gns3.com/software [Accessed 18 August 2018].

Iguchi-Cartigny, J., 2014. *Netkit-NG Homepage / Netkit-NG.* [Online] Available at: https://netkit-ng.github.io/ [Accessed 23 July 2018].

Josephson, J. R. & Josephson, S. G., 1994. *Abductive Inference: Computation, Philosophy, Technology.* Cambridge: Cambridge University Press.

Li, J., Ou, X. & Rajagopalan, R., 2010. Uncertainty and Risk Management in Cyber Situational Awareness. In: S. Jajodia, P. S. V. Liu & C. Wang, eds. *Cyber Situational Awareness - Issues and Research.* New York: Springer, pp. 51-68.

Mateski, M. et al., 2012. *Cyber Threat Metrics,* Albuquerque: Sandia National Laboratories.

Mell, P. et al., 2012. *CAESARS Framework Extension: An Enterprise Continuous Monitoring Technical Reference Model (Second Draft),* Gaithersburg, MD, US: s.n.

Nathans, D., 2015. *Designing and Building a Security Operations Center.* Massachusetts:

Syngress.

National Institute of Standards and Technology, 2011. *NVD - CVE-2011-3192.* [Online] Available at: https://nvd.nist.gov/vuln/detail/CVE-2011-3192 [Accessed 18 August 2018].

National Institute of Standards and Technology, 2018. *NVD - Search and Statistics.* [Online] Available at: https://nvd.nist.gov/vuln/search [Accessed 6 August 2018].

NETKIT, 2009. *Features - Netkit Wiki.* [Online] Available at: http://wiki.netkit.org/index.php/Features [Accessed 30 June 2018].

NETKIT, 2016. *Download Official - Netkit Wiki.* [Online] Available at: http://wiki.netkit.org/index.php/Download_Official [Accessed 17 July 2018].

Pendleton, M., Garcia-Lebron, R., Cho, J.-H. & Xu, S., 2016. A Survey on Systems Security Metrics. *ACM Computing Survey,* 49(4), p. 62.

Raulerson, E. L., Hopkinson, K. M. & Laviers, K. R., 2015. A framework to facilitate cyber defense situational awareness modeled in an emulated virtual machine testbed. *The Journal of Defense Modeling and Simulation,* 12(3), pp. 229-239.

Sawilla, R. & Burrell, C., 2010. *Metrics-based Computer Network Defence Decision Support.* Ottawa, DEFENCE RESEARCH AND DEVELOPMENT CANADA OTTAWA (ONTARIO).

Symantec Corporation, 2017. *Ransom.Wannacry | Symantec.* [Online] Available at: https://www.symantec.com/security-center/writeup/2017-051310-3522-99 [Accessed 2 August 2018].

Tadda, G. P. & Salerno, J. S., 2010. Overview of Cyber Situation Awareness. In: S. Jajodia, P. S. V. Liu & C. Wang, eds. *Cyber Situational Awareness - Issues and Research.* New York: Springer, pp. 15-35.

Wilson, J., 2014. *Essentials of Business Research: A Guide to Doing Your Research Project.* Second ed. s.l.:SAGE.

Zhai, J. & Wang, K., 2011. *Design and Implementation of Dynamic Virtual Network.* Harbin, International Conference on Electronic & Mechanical Engineering and Information Technology.

Appendix A: Consent Form

PROJECT TITLE: Model proposal for analysing the threat level and the cost benefit of a possible security solution oriented to a TCP/IP network

NAME OF RESEARCHER: Ayner Antonio Pérez Tito

I confirm that I have read and understood the provided Participant Information Leaflet (PIL) for the above project, and that I have had the opportunity to ask any questions about the research that I may have.

Further, I have been given a copy of the PIL which I may keep for my records.

I agree to take part in the above study and am willing to have my involvement in the interview noted.

Further, I have additionally agreed to have the interview electronically recorded.

I understand that my information will be held and processed to be used anonymously for internal publication for an MSc project, to be submitted for assessment for an MSc degree. I also understand that such anonymous data may be used for future research, including that for publication.

I understand that my participation is voluntary and that I am free to withdraw at any time up to the submission of the dissertation without giving any reason and without being penalised or disadvantaged in any way.

[**# ..... Item to be deleted if not appropriate or if permission is not granted by the Participant**]

Name of participant ..... Date .....

Signature .....

Appendix B: Participant Information Leaflet (PIL)

This sheet seeks to provide information, and advice, with respect to an individual's participation in support of the specified research project:

1. The project is entitled '*Model proposal for analysing the threat level and the cost benefit of a possible security solution oriented to a TCP/IP network*, and will consider the *Cyber Security* aspects related to this subject;
2. This research is being conducted by *Ayner Antonio Pérez Tito* in support of his studies for an MSc at the University of Warwick, and this research is self-funded by the student;
3. The research is being supervised by *Harjinder Singh Lallie*, *HL@warwick.ac.uk*, who is a member of the teaching staff at the University.
4. Participation in this research is totally voluntary, and assurances are given to the effect that no negative consequences will arise from refusal to participate, from limiting participation, or from withdrawing (prior to dissertation submission) input that arose from any earlier participation in the research project;
5. Each individual is advised to fully consider, with others if necessary and prior to participation, any disadvantages, side effects, risks and/or discomforts that may arise from participation in this research;
6. Unless specifically agreed otherwise, all information will be carefully made anonymous, and all the data on such original sources will be held as confidential and will not be distributed to others;
7. The resulting dissertation, with anonymous data, will be reviewed by a University teaching staff member and/or a University appointed external assessor, by the University moderators, and by external examiners;
8. Whilst an MSc Dissertation does not pass into the public domain, it is possible that the dissertation (with its data) may be used as a source for future research, including research work for publication;
9. Whilst summarised/ analysed data may be used in future research and/ or publications, your individual data responses will be retained only until the student completes their course and then destroyed.

This research has been favourably reviewed by the University's Biomedical and Scientific Research Ethics Committee, Approval Reference: XXXXXXXX, dated: XX xxxxxx 2018. Dissatisfaction with the conduct of this research may be referred to the person below, who is a senior University of Warwick official entirely independent of this study:
Head of Research Governance, Research & Impact Services, University House, University of Warwick, Coventry, CV4 8UW;
Tel: 024 76 522746; Email: researchgovernance@warwick.ac.uk

- Completion of all or part of a survey by a participant will be deemed as permission to use the data provided within the dissertation.

Appendix C: Ethical Approval confirmation

Ethical Approval confirmation

wmg-ftmasters@warwick.ac.uk

Wed 25/07/2018 08:57

To: Pérez Tito, Ayner <A.Perez-Tito@warwick.ac.uk>;

Cc: Lallie, Harjinder <HL@warwick.ac.uk>;

Dear Mr Pérez Tito,
Warwick University ID Number: 1793561

Thank you for submitting your Supervisor's Delegated Approval form to the FT Course Office for the project: Model Proposal for Analysing the Threat Level and the Cost Benefit of a Possible Security Solution Oriented to a TCP/IP Network.

Your reference number is REGO-2017-WMG-0877.

You now have the appropriate approval in place to begin your study.

Please ensure you insert a copy of this email into the appendices of your project.

Best Wishes

Laura Dobson
WMG Full-Time MSc Course Office
wmgcourseoffice@warwick.ac.uk
go.warwick.ac.uk/wmgftmsc
+44(0)24 7657 4206

Appendix D: Metrics defined by Pendleton, et al.

The following is a summary of the different metrics mentioned by Pendleton, et al. (2016) and a description for each of them.

- **Vulnerability V(t):** Function related to vulnerabilities at a "t" point in time to tell the level of vulnerability in the system.
    - **User vulnerabilities:** Vulnerabilities based on user's errors or lack of knowledge. These vulnerabilities are related to topics like phishing or malware susceptibility and password issues.
    - **Interface-Induced Vulnerabilities:** Vulnerabilities linked to the interface used by a client to interact with a server. These include the port used for an application and the protocols and processes used to transmit and receive data.
    - **Software Vulnerabilities:** It has three categories of measurement:
        - **Temporal attributes of vulnerabilities:** It includes the evolution of vulnerabilities in a software during its history and the average time for a patch to be available and installed.
        - **The severity of Individual Software Vulnerabilities:** To assess the severity there are different scoring systems like the Common Vulnerability Scoring System (CVSS) and the Common Weakness Scoring System (CWSS).
        - **The severity of a Collection of Vulnerabilities:** It measures the effect of multiple vulnerabilities being on the same system together. For analysing the vulnerabilities there are multiple approaches: attack graphs, Bayesian Networks, attack trees and privilege trees. The possible metrics could be Deterministic (based on attack graphs and the effort needed to mitigate or exploit a vulnerability) or Probabilistic (based on the likelihood of exploitation and the worst-case scenario)
- **Defence D(t):** Function related to the defences present at a particular time. It measures the strength of the different defence mechanisms:

  - **Preventive Defences:** Mechanisms like blacklisting, Data Execution Prevention, and Control-Flow Integrity aimed to block attacks. Each mechanism has its own metrics.
  - **Reactive Defences:** Includes Intrusion Detection Systems and antiviruses. It includes several metrics for Monitoring, and also Detection.
  - **Proactive Defences:** Includes Address Space Layout Randomization (ASLR) and Moving Target Defense (MTD), two mechanisms that allow the network to change regularly and make difficult the attack process. Each of them has a set of metrics.
  - **Overall Defences:** Comprehends two metrics: Penetration Resistance (level of effort to penetrate a system) and Network Diversity (average effort to compromise a target based on the length of the attack path).
- **Attack A(t):** Function related to the attacks performed at a time "t" and measures the strength of the attacks against the system.
  - **Zero-Day attacks:** Involves two metrics: Time between the attack being launched and the moment it is identified and disclosed to the public. Another metric is the number of victims of the attack.
  - **Targeted attacks:** Attacks directed to a particular target and involve how sophisticated the delivery method is and the complexity of the attack
  - **Botnets:** Involves different metrics to measure how harmful a botnet attack could be
  - **Malware spreading:** Defined by the infection rate metric: the time for a malware to spread from one computer to several.
  - **Attack Evasion:** Metrics to measure how successful an attack is for evading security defences. It involves metrics for measuring Adversarial Machine Learning Attacks and Obfuscation attacks.
- **Situation S(t):** It is a function that depends on V(t), A(t) and D(t) and reflects the actual state of the system in a particular point in time. However, it comprehends the following metrics for measuring:
  - **Security State:** Metrics oriented to measure the state of the system based on the data (Data-Driven State Metrics) and based on the outcome of the interaction between the attacks and the defence mechanisms (Model-

Driven Metrics).

- **Security Incidents:** Addresses the severity and impact of previous incidents inside the elements in the system.
- **Security investment:** Considers the budget and overall investment in acquiring and maintaining security defences

Appendix E: Metrics defined by Tadda & Salerno and Duggan, et al.

1. Categories mentioned by Tadda & Salerno (2010)

For a better understanding of the categories proposed by Tadda & Salerno (2010), the following example is used: if some metric states that several attack attempts were performed on a server, the following questions would be:

a) **Confidence:** How much confidence the manager has that the number of attacks is well measured and identified? By going further, metrics can be measured in terms of how reliable they can be. This is also analysed as trustfulness by Barford et al. (2010). The confidence metrics inside this group are numerical formulas:
   a. **Recall:** Correct detections vs the total of known attack tracks (fully recognized attack tracks)
   b. **Precision:** Correct detections vs detected attack tracks (hypothetical attack tracks under evaluation)
   c. **Fragmentation:** Number of fragments (attack track that was part of a bigger attack track) vs detected attack tracks
   d. **Mis-association:** All other detected tracks vs detected attack tracks

b) **Purity:** How much of the evidence related to each attack has been correctly identified as part of it? This item can also be seen as how well the system or manager was able to recognize the full extension of an attack.

c) **Cost-Utility:** How much is the impact of the attacks on the system? By including weights to the metrics, a manager can emphasize which ones have more impact on the system. For example, the attack score is one of those metrics (later known as Activities of Interest or AOI)

d) **Timeliness:** How quick the system or manager responds and mitigates the attacks? This parameter considers the time lapse between a detected attack and the response to counteract its effect.

2. Categories mentioned by Dugan, et al. (2007)

As mentioned in the literature review, in this research a metric categorization is proposed

highlighting the relevance of visualizing the metrics.

THREAT LEVEL	THREAT PROFILE						
	COMMITMENT			RESOURCES			
	INTENSITY	STEALTH	TIME	TECHNICAL PERSONNEL	KNOWLEDGE		ACCESS
					CYBER	KINETIC	
1	H	H	Years to Decades	Hundreds	H	H	H
2	H	H	Years to Decades	Tens of Tens	M	H	M
3	H	H	Months to Years	Tens of Tens	H	M	M
4	M	H	Weeks to Months	Tens	H	M	M
5	H	M	Weeks to Months	Tens	M	M	M
6	M	M	Weeks to Months	Ones	M	M	L
7	M	M	Months to Years	Tens	L	L	L
8	L	L	Days to Weeks	Ones	L	L	L

Figure 1: Generic Threat Matrix (Duggan, et al., 2007)

The matrix addresses a level of threat for each metric in terms of capability, making it possible to compare the metrics. It defines two main threat attributes: Commitment (willingness of a threat to achieving its purpose) and Resources (information available to the threat for achieving its purpose).

Appendix F: SAEM analysis model

The process of the SAEM analysis model to asses the network is as follows:

a) **Multi-Attribute Risk Assessment:** It consists of identifying the outcome attributes like lost revenue, government penalties, reputation, etc. From there each outcome is assigned with possible units of their effect (money, hours, etc), as well as the frequency or probability the outcome can occur. An example of these steps is shown in Figure 1. Next is ranking each outcome according to the concerns of the company. In Figure 2, an example of a ranking from 1 to 100 is presented for each attribute where 100 is the biggest concern of the company, and from there a method of normalizing the weight is used. Finally, a threat index is made based on possible risks the network can face. This index normalizes the values by using a function, so the unit assigned to each value can be compared. The threat index is then used by the SAEM.

b) **Benefits Assessment:** This is the first step of the SAEM. It determines the mitigation provided by the technology. Each security technology is categorized according to the effect they have on the risk. An example of categorizing is the defence-in-depth model, which defines protection, detection and recovery mechanisms. If a technology falls in more than one category, it should be considered in both. The next step is defining which threat is mitigated by which technology. The last step is quantifying the level of effectiveness of each mitigation process. Effectiveness is not a precise value and only estimates can be used. The estimates are obtained by the experience of the network or security managers.

c) **Threat Index Evaluation:** For each possible security technology, the possible risks that could be mitigated by them are grouped. Then, an estimation of the risk reduction is performed by considering the threat index obtained in the first step and a new threat index analysis made by considering the possible technology to be implemented and how much it can lower the frequency of the threat. By doing so, it is possible to estimate the risk reduction made by adding a new piece of hardware or software into the network.

Threats (Estimated Attacks/year)		Lost Revenue ($$)	Reputation (0- 6 Scale)	Lost Productivity (hours)	Regulatory Penalties (0- 6 Scale)
Scanning (10,220/yr)	Low	0	1	.25	0
	Exp	0	1	.5	0
	High	1,000	4	1	0
Procedural Violation (4,380/yr)	Low	0	0	0	0
	Exp	0	1	2	0
	High	12,000	4	40	3
Browsing (2,920/yr)	Low	0	0	0	0
	Exp	0	1	0	0
	High	0	4	8	0

Figure 1: Extract of an example of steps 1 and 2 of a Multi-Attribute Risk Assessment (Butler, 2002)

Attributes	Ranks	Weights (W)
Lost Productivity	100	.42
Reputation	80	.33
Regulatory Penalties	40	.17
Lost Revenue	20	.08

Figure 2 Extract of an example of step 3 of a Multi-Attribute Risk Assessment (Butler, 2002)

d) **Security Architecture Coverage:** Another attribute to consider, besides the effectiveness of a solution, is the principle that "there should be at least one mitigation strategy for each risk" (Butler, 2002). Even if one technology might prove to be more effective in the overall threat mitigation another technology could cover any unmanaged threat and therefore this last one should be taken as a priority. The same happens if there is only ONE security mechanism. This might influence the technology to adopt.

e) **Cost:** This last option involves different types of cost: purchase and installation, employees training, operation and management costs, licensing fees. Because of the previous step, only a couple of options are considered so the time needed to evaluate each technology's cost is reduced. By analysing all the costs involved, a final estimate can be obtained to compare each technology.

Appendix G: Options for network emulators

1. GNS3

GNS3 is a network emulator used for running the operative system of communication systems like routers. Allows the user to deploy software from different manufacturers and emulate their performance. Moreover, it is possible to connect real network environments into the virtual network (Galaxy Technologies, LLC, 2018).

This software is free to use. However, the images of routers and any other terminal to be used requires licensing. As an example, GNS3 apologizes for not being able to provide with Cisco images because of legal requirements (Bombal, 2017).

2. IOU-Web/EVE

Andrea Dainese created a portable tool for emulating networks called *IOU-Web*. From her work, several solutions were developed including Unified Networking Lab or *UNETLAB*, *UNETLABv2* that is not available to the public, and Emulated Virtual Environment (EVE) (Dainese, 2017).

EVE Next Generation (EVE-NG) is available in different formats: Community and Professional, being the first one free of use. However, it runs several hypervisors and it is recommended to use a dedicated server for running the environment (EVE-NG, 2017).

3. Netkit-NG

Netkit NG is based on the original *Netkit* project (Iguchi-Cartigny, 2014). Both projects are based on Debian and use a common filesystem as a base to construct the elements in the network. It can emulate and interconnect several elements as part of a network, allowing the user to implement several services inside each terminal according to the needs of the user.

At the moment of its release, it was using a supported version of *Debian*. However, at this point in time, the version is no longer supported. The outcome is that installing new services becomes more complex and requires a higher level of knowledge by the user.

Appendix H: Configuration files for the experiment

In Figure 1 and 2, the directory structure of the project is presented:

```
computer@computer-VirtualBox:~/experiment$ ls -R
.:
DNS                   FW1.startup       lab.conf      Workstation1.startup
DNS.startup           FW2               lab.dep       Workstation2
ExternalUser          FW2.startup       Web           Workstation2.startup
ExternalUser.startup  Internet          Web.startup
FW1                   Internet.startup  Workstation1
```

Figure 1: Contents in the main directory of the experiment

```
./DNS:
etc

./DNS/etc:
dnsmasq.conf  dnsmasq_hosts.conf  hosts
resolv.conf

./ExternalUser:

./FW1:
etc

./FW1/etc:
hosts  resolv.conf

./FW2:
etc

./FW2/etc:
hosts  resolv.conf

./Internet:
```

```
./Web:
etc

./Web/etc:
hosts  resolv.conf

./Workstation1:
etc

./Workstation1/etc:
hosts  resolv.conf

./Workstation2:
etc

./Workstation2/etc:

hosts  resolv.conf
```

Figure 2: Contents in subdirectories of the experiment

In the following figures, the contents of each file are presented. Note that lab.dep does not have any content inside.

```
#Assigning IPs to the interfaces
ifconfig eth0 172.16.1.3/24 up

#Default route to outside the network
route add default gw 172.16.1.1

#Start the services
service dnsmasq start
```

Figure 3: Contents in DNS.startup

```
#Assigning IPs to the interfaces
ifconfig eth0 10.2.0.2/24 up

#Default route to outside the network
route add default gw 10.2.0.1
```

Figure 4: Contents in ExternalUser.startup

```
#Assigning IPs to the interfaces
ifconfig eth0 10.1.0.1/30 up
ifconfig eth1 10.0.0.2/24 up
ifconfig eth2 172.16.3.1/24 up

#Adding the routes to other networks
route add default gw 10.0.0.1
ip route add 172.16.1.0/24 via 10.1.0.2
ip route add 192.168.0.0/24 via 10.1.0.2

#Network Address Translation to Internet (SNAT)
iptables -t nat -A POSTROUTING -o eth1 -s 192.168.0.0/24 -j SNAT --to-source
10.0.0.26-10.0.0.30 #Workstations
iptables -t nat -A POSTROUTING -o eth1 -s 172.16.1.0/24 -j SNAT --to-source
10.0.0.17 #InnerDMZ
iptables -t nat -A POSTROUTING -o eth1 -s 172.16.3.0/24 -j SNAT --to-source
10.0.0.19 #OuterDMZ
iptables -t nat -A POSTROUTING -o eth1 -s 10.1.0.0/30 -j SNAT --to-source 10.0.0.20
#Management

#Network Address Translation from nodes on Internet to the network (DNAT)
iptables -t nat -A PREROUTING -i eth1 -d 10.0.0.34 -j DNAT --to-destination
172.16.1.3 #DNS
iptables -t nat -A PREROUTING -i eth1 -d 10.0.0.38 -j DNAT --to-destination
172.16.3.2 #Web Server
```

Figure 5: Contents in FW1.startup

```
#Assigning IPs to the interfaces
ifconfig eth0 192.168.0.1/24 up
ifconfig eth1 10.1.0.2/30 up
ifconfig eth2 172.16.1.1/24 up

#Adding the routes to other networks
route add default gw 10.1.0.1
```

Figure 6: Contents in FW2.startup

```
#Assigning IPs to the interfaces
ifconfig eth0 10.0.0.1/30 up
ifconfig eth1 10.2.0.1/24 up

#Adding the routes to other networks
ip route add 10.0.0.16/28 via 10.0.0.2
ip route add 10.0.0.32/28 via 10.0.0.2

iptables -t nat -A POSTROUTING -o eth2 -j MASQUERADE
```

Figure 7: Contents in Internet.startup

```
LAB_VERSION=1.0
LAB_AUTHOR=1793561
LAB_DESCRIPTION="Experiment for testing the model"

DNS[mem]=512
FW1[mem]=512
FW2[mem]=512
Web[mem]=512

Workstation1[0]=LANWORK
Workstation2[0]=LANWORK
FW2[0]=LANWORK
FW2[1]=P2PFW
FW1[0]=P2PFW
FW2[2]=INNERDMZ
DNS[0]=INNERDMZ
FW1[2]=OUTERDMZ
Web[0]=OUTERDMZ
FW1[1]=INTERNET
Internet[0]=INTERNET
Internet[1]=EXTERNALUSER
ExternalUser[0]=EXTERNALUSER
Internet[2]=tap,10.10.0.1,10.10.0.2
```

Figure 8: Contents in lab.conf

```
#Assigning IPs to the interfaces
ifconfig eth0 172.16.3.2/24 up

#Default route to outside the network
route add default gw 172.16.3.1

#Start the services
service apache2 start
```

Figure 9: Contents in Web.startup

```
#Assigning IPs to the interfaces
ifconfig eth0 192.168.0.2/24 up

#Default route to outside the network
route add default gw 192.168.0.1
```

Figure 10: Contents in Workstation1.startup

```
#Assigning IPs to the interfaces
ifconfig eth0 192.168.0.3/24 up

#Default route to outside the network
route add default gw 192.168.0.1
```

Figure 11: Contents in Workstation2.startup

The comments in dnsmasq.conf have been erased for a clearer view of the commands needed. However, it is recommended to check the comments for further understanding.

```
domain-needed
bogus-priv
no-resolv
no-poll
server=8.8.8.8
local=/company.test/
interface=eth0
no-dhcp-interface=eth0
no-hosts
addn-hosts=/etc/dnsmasq_hosts.conf
expand-hosts
domain=company.test
```

Figure 12: Contents in DNS/etc/dnsmasq.conf

```
192.168.0.2 workstation1.company.test
192.168.0.3 workstation2.company.test
172.16.1.3 dns.company.test
172.16.3.2 web.company.test
```

Figure 13: Contents in DNS/etc/dnsmasq_hosts.conf

```
127.0.0.1    localhost
127.0.1.1    DNS.company.test DNS
```

Figure 14: Contents in DNS/etc/hosts

The comments in resolv.conf have been erased on each of the files for a clearer view.

```
nameserver 172.16.1.3
```

Figure 15: Contents in DNS/etc/resolv.conf

```
127.0.0.1    localhost
127.0.1.1    FW1.company.test FW1
```

Figure 16: Contents in FW1/etc/ hosts

```
nameserver 172.16.1.3
```

Figure 17: Contents in FW1/etc/resolv.conf

```
127.0.0.1    localhost
127.0.1.1    FW2.company.test FW2
```

Figure 18: Contents in FW2/etc/ hosts

```
nameserver 172.16.1.3
```

Figure 19: Contents in FW2/etc/resolv.conf

```
127.0.0.1    localhost
127.0.1.1    Web.company.test Web
```

Figure 20: Contents in Web/etc/ hosts

```
nameserver 172.16.1.3
```

Figure 21: Contents in Web/etc/resolv.conf

```
127.0.0.1    localhost
127.0.1.1    Workstation1.company.test Workstation1
```

Figure 22: Contents in Workstation1/etc/ hosts

```
nameserver 172.16.1.3
```

Figure 23: Contents in Workstation1/etc/resolv.conf

```
127.0.0.1    localhost
127.0.1.1    Workstation2.company.test Workstation2
```

Figure 24: Contents in Workstation2/etc/ hosts

```
nameserver 172.16.1.3
```

Figure 25: Contents in Workstation2/etc/resolv.conf