Towards a Critical Review of Cybersecurity Risks in Anti-Poaching Systems in South Africa

by

Christelle Steyn

Thesis presented in fulfilment of the requirements for the degree of Master of Arts in Socio-Informatics in the Faculty of Arts and Social Sciences at the University of Stellenbosch

Supervisor: D.N. Blaauw

December 2022

Declaration

By submitting this thesis electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third-party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Abstract

Anti-poaching operations increasingly make use of a wide variety of technology for intelligence and communications. These technologies introduce risk, and they need to be secured to provide greater protection to the information and people involved in anti-poaching, ultimately protecting the animals better. A hypothetical network of anti-poaching technologies was simulated in GNS3, consisting of various field devices and a main control room. A Kali Linux machine was connected to the network and played the role of an attacker. Several cyber-attacks were carried out, to show the risks inherent to such a network. These attacks were then mitigated via system configurations. Further risks were identified in the literature. Using the STRIDE and DREAD threat models, the risks to an anti-poaching network were classified and calculated. The Denial of Service and Elevation of Privilege classes posed the most risk to the system. Mitigations to general network threats and those from the simulation are mentioned. Authentication for such a system was investigated, as improper authentication practices were deemed a risk. Recommendations made, include the proper configuration of network devices, the use of anti-virus, firewalls, and intrusion detection systems, as well as having an external audit performed annually. Multi-factor authentication, with a password/fingerprint combination, is recommended.

Uittreksel

Teen-stropery aksies maak al hoe meer gebruik van ‘n wye verskeidenheid tegnologie vir intelligensie en kommunikasie. Die tegnologie wat gebruik word bevat inherente risikos, en moet dus beveilig word om die sekuriteit van die inligting en mense betrokke by teen-stropery te verhoog, en daardeur die diere verder te beskerm. ‘n Hipotetiese netwerk van teen-stropery tegnologieë was in GNS3 gesimuleer, en het bestaan uit ‘n verskeidenheid veldtoestelle en ‘n hoof kontrolekamer. ‘n Kali Linux masjien was aan die netwerk gekoppel en het die rol van ‘n aanvaller gespeel. Verskeie kuber-aanvalle is gedoen om te wys watter risikos daar in die netwerk bestaan. Hierdie risikos is aangespreek deur spesifike konfigurasies om die effektiwiteit te bepaal. Verdere risikos is in die literatuur geidentifiseer. Deur gebruik te maak van die STRIDE en DREAD bedreigingsmodelle, was die risikos in ‘n teen-stropery netwerk geklassifiseer en bereken. Die “Denial of Service” en “Elevation of Privilege” klasse het die meeste risiko vir die sisteem ingehou. Voorgestelde beheermaatreëls vir algemene netwerk bedreigings en die van die simulasie word genoem. Gebruiker verifikasie vir sulke sisteme was nagevors, want onbehoorlike verifikasie gebruike word ook as ‘n risiko beskou. Aanbevelings wat gemaak word sluit die behoorlike konfigurasie van die netwerk, die gebruik van teen-virus sagteware, “firewalls” en “intrusion detection systems” in, sowel as om jaarliks ‘n eksterne oudit te laat uitvoer. Verder, word veel-faktor verifikasie, met ‘n wagwoord/vingerafdruk kombinasie, aanbeveel

Acknowledgements

I could not have done this masters without the assistance and support of family and friends. I would like to extend my thanks and gratitude to the following people.

To IS, for his PhD worthy comments and support.

To DS, for her continued love and support.

To NS, for tolerating all my academic ramblings, so that I could order my thoughts. And for always accommodating me.

To AS and AAS, for their support and continued interest in this thesis.

To MS, JS, and SS, just for being there.

To RvL, for thesis advice, coffees, and Discord sessions.

To BM and PB, for the social breaks they provided, be it boardgames or videogames.

To NN, my Cisco network guru, for helping me to set up my simulation.

To DNB, a superior supervisor, for his continuous guidance and support, for his answers to my myriad of questions at the start, and for believing in me and this Master’s.

Table of Contents

List of Acronyms

Acronym	Full Text
ARP	Address Resolution Protocol
BPDU	Bridge Protocol Data Unit
CAM	Content Addressable Memory
CCF	Connected Conservation Foundation
CIA	Confidentiality, Integrity, and Availability
DoS	Denial of Service
DHCP	Dynamic Host Configuration Protocol
DMZ	Demilitarised Zone
DTP	Dynamic Trunking Protocol
GNS3	Graphical Network Simulator 3
GPS	Global Position System
IoT	Internet of Things
IP	Internet Protocol
ITU	International Telecommunications Union
JOC	Joint Operations Centre
L2	Layer 2
MAC	Medium Access Control
MitM	Man in the Middle
OS	Operating System
OWASP	Open Web Application Security Project
PC	Personal Computer
PIN	Personal Identification Number
QEMU	Quick Emulator
RFID	Radio Frequency Identification
SANParks	South African National Parks
STP	Spanning Tree Protocol
TCP	Transmission Control Protocol
VLAN	Virtual Local Area Network
VM	Virtual Machine
WWF	World Wide Fund for Nature

Glossary

Anti-poaching – any method or measure employed to counter poaching

ARP – A protocol used to convert IP addresses into MAC addresses

BPDU – A unit containing information enabling it to become the root of the STP

CAM – A memory table constructed by the system to store MAC addresses, ports, and VLAN IDs

CIA – The traditional triad of confidentiality, integrity and availability that informs some cybersecurity frameworks

Cyber-attack – When one party maliciously breaches the security of another party's system

Cybersecurity – The people, practices and technologies used to protect systems and information from cyber-attacks

DoS – Attacks where the usual operation of a system is disrupted by an overload of network traffic

DHCP – A protocol used to assign DNS servers and IP addresses to new connections to the network

DMZ – A subnetwork that is positioned between a private network and the Internet, and aids in protecting against untrusted network traffic

DTP – A protocol that that detects trunk links between switches

Hacking – Relates to malicious actions performed to compromise networks and digital devices

IoT – Refers to the use of physical devices such as sensors and tags that are connected to the Internet

IP – A protocol that determines the format with which data is sent across a network or the Internet

L2 – The data link layer in network protocol design, it is used to transfer data between adjoining network nodes

MAC – Also called a hardware address, it refers to an alphanumeric string used to identify devices on a network

MitM – An attack where an attacker inserts themselves into the “middle” of communications on a network to intercept data

Poaching – Any extractive use of wildlife that is considered illegal by the state

Risk – The perception created by the probability and consequences of a threat occurring

STP – A protocol implemented on a switch to form a loop free topology based on a looped physical topology

TCP – A protocol designed to transfer packets across the internet via a three-way handshake

Threat – Any malicious action that can intercept, damage, or disrupt data or systems

Threat modelling – The identification and understanding of threats and applicable mitigations with critical systems in mind

VLAN – When a subset of devices sharing a physical network is grouped into an overlay network to separate network traffic

VM – A resource that emulates a computer and its components to run programmes, via software instead of physical parts

List of Figures

List of Tables

List of Equations

1. Introduction

1.1. Background Information

Poaching has become increasingly problematic for wildlife conservation (Cooney, *et al.*, 2016). In the South African context, rhinos are the chief focus of poachers. According to Save The Rhino (2020), the current rhino poaching crisis started in 2008, and reached a peak in 2015, having increased by 9,000% from 2007 to 2014. The crisis is not over, and current anti-poaching strategies may be insufficient (Yang, *et al.*, 2014). Therefore, conservationists and anti-poaching units need all the tools and techniques they can acquire to help save wildlife.

This crisis requires measured intervention. This is where reliable technology enters the picture. Lancaster (2018) cites Colby Loucks, a WWF director at the time, as saying that his hope for the future is that "the technology [in conservation] has become as common as getting a uniform, boots or a radio." Thevar and Bhanot (2021) state that "With the advent of technology, however, there have been substantial improvements in anti-poaching efforts across the globe."

As anti-poaching units rely on their ground data and communication networks, it is imperative that these networks are well set up and securely authenticated, but also well protected from digital attacks. In Thevar and Bhanot's (2021) research, they emphasize that there has been a dramatic improvement in anti-poaching technology over the years, but they state that "the misuse of technology is a looming concern". According to the authors, the advancement in technology for the rangers is mirrored for the poachers. It is increasingly likely that poachers can potentially intercept data feeds and signals from trackers placed on animals or patrol units. The data stored and sent by network components and Internet of Things (IoT) devices, are targets for cyber-attacks and can be vulnerable if not secured properly. The authors therefore make a call for the exploration of more ways in which servers and tracking tags can be protected (Thevar & Bhanot, 2021).

It is acknowledged that there is more to stopping poaching than just the technology and the cybersecurity aspect. It remains important to note that "While technology has improved monitoring and strategy efforts, poaching will remain a problem as long as there is a demand for wildlife products" (Thevar & Bhanot, 2021), and according to one conservationist, as cited by Lancaster (2018), "no matter how advanced the technology, the human element will always lead the way." Ball, *et al.*, (2018) also argue that one should strive "to get the basics of rhino protection in place before the addition of more sophisticated technological layers." This is mentioned to provide context to the poaching problem, but the research presented here will solely focus on the cybersecurity of anti-poaching systems and will not make comment on other strategies and the way forward for anti-poaching operations in general.

1.2. Introduction to Poaching and Anti-Poaching

The poaching of animals presents itself in many different forms and for many different purposes. Thus, there are also many ways to define poaching. "The illegal hunting of wild animals to trade their meat or body parts for various purposes" is characterized as poaching by Thevar and Bhanot (2021), and it can lead to the extinction of species. This definition however glosses over the aim of harvesting wildlife for subsistence purposes. As what is "illegal" varies from country to country, the following definition by Duffy (1999) for poaching may provide a better framework: poaching is "any extractive use of wildlife that is considered illegal by the state."

Poaching has been a recurring problem throughout history. However, over the last few decades, especially since 2008 (Save The Rhino, 2020), there has been an increase in poaching and its intensity in protected areas and on protected animals, such as rhinos and elephants in national parks.

The rise in the illegal wildlife trade, poverty and the allure of riches can be attributed to the renewal in poaching efforts (Cooney, *et al*., 2016 & Massawe, *et al*., 2017). The loss of wildlife, biodiversity, and natural beauty, coupled with a sharper awareness of ecosystem collapse and climate change amongst conservationists, have increasingly placed focus on the poaching problem in national parks, wildlife reserves and private game farms. A concentrated effort needs to be made to eliminate, or greatly reduce, the poaching problem, and this requires effective anti-poaching strategies.

Anti-poaching refers to any measures, methods or operations used to curb and counter the relevant state's definition of poaching. Anti-poaching works in opposition to poaching. Most anti-poaching operations involve specialised ranger units that work alongside a headquarter, such as the Joint Operations Centres (JOCs) used by the South African National Parks (SANParks). As is the case with SANParks, technology facilitates the communications between the JOC and its patrol units, and technology provides the monitoring equipment and resources to the JOC. To optimise the use of both human and technology resources in anti-poaching strategy and tactics, the location of animals is an important factor, with this information often being managed using technology. Therefore, there are many actors and technologies that can be involved in a digitised anti-poaching system. To limit the dangerous risks involved in anti-poaching and to protect access to critical information, such systems need to be highly secure. Applying cybersecurity principles to these systems would therefore be best practice.

Based on this, one can classify a typical anti-poaching system as a socio-technical system, featuring interoperable social aspects and integrated technologies. A hypothetical instance of such a system will be analysed, and an attempt will be made to develop a threat model with which the risks to such a system can be calculated, and in turn be used to determine mitigations.

1.3. Introduction to Cybersecurity

Cybersecurity involves the protection of assets or the security of information in a cyber environment, says the International Telecommunication Union (ITU) (International Telecommunication Union, 2022). Traditionally concerned with ensuring the triad of data confidentiality, data integrity and data availability, cybersecurity has needed to expand in recent decades to accommodate the rise in socio-technical systems (Samonas & Coss, 2014).

While poaching has been increasing, so too has the need for cybersecurity in digitised anti-poaching systems. The proliferation of technology and connectivity in the modern day, accompanied by old and new cyber risks, have generated a rising demand for what cybersecurity can offer (Amin, 2017).

> Cybersecurity is the collection of tools, policies … [and] safeguards … that can be used to protect the cyber environment and organization[‘s] … assets. [These] include connected computing devices, personnel, infrastructure, applications, services, telecommunications systems, and the totality of transmitted and/or stored information in the cyber environment. Cybersecurity strives to ensure the attainment and maintenance of the security properties of the organization[‘s] … assets. (ITU, 2022)

All the components of an organization’s network therefore need to be taken into consideration when determining and responding to cybersecurity threats, per the ITU’s definition above. Cybersecurity focuses on both the human and technology elements in a system (making it applicable to socio-technical systems), as humans divulging classified information is just as much a threat to the system as a poacher being able to attack the system and steal data.

However, there is one limitation to the ITU's definition above. The ITU's definition is devoid of the concept of "risk" – a concept deemed fundamental to cybersecurity (Ciampa, 2015 & Sotnikov, 2022). "If you work in security, you are in the risk management business", is a statement made by Sotnikov (2022), that emphasises the centrality of risk in cybersecurity.

Sotnikov (2022) provides the following definition for cybersecurity: "Cybersecurity is all about understanding, managing, controlling and mitigating risk to your organization's critical assets." Therefore, while the ITU's definition is very broad and describes the cyber environment in detail, Sotnikov's definition revolves around the concept of the risk posed by threats and how managing this risk leads to the securing of an organisation's assets. A combination of Sotnikov's and the ITU's definitions can provide a picture of risk management with regards to all the aspects mentioned by the ITU.

1.4. Problem Statement

As the battle against poaching ramps up and technology evolves and becomes more accessible, many anti-poaching units have started to incorporate various technologies into their arsenal, such as tags, cameras, and advanced monitoring and communication equipment. This makes it increasingly necessary to address newly introduced cybersecurity risks. As Schoenfield (2015) notes, attackers have not only become more, but they are also achieving higher levels of sophistication. Therefore, formulating risk mitigation strategies for anti-poaching systems should have a high priority, as these systems are sensitive in nature and lend themselves to being exploited by criminals.

Due to the confidential nature of anti-poaching systems, it is of critical importance that the systems and personnel be protected, to in turn aid in protecting dwindling wildlife populations.

A literature review will be presented on the use of technology in anti-poaching strategies, as well as some preliminary cybersecurity, authentication, and risk considerations.

While studies have been conducted on the use of various technologies in anti-poaching systems and some do place their focus in the context of South African wildlife, very few address the specific cybersecurity concerns of such systems.

It is also important to note that mention is made of criminal syndicates in poaching discussions (ZAPWing, 2022). While a typical poaching recruit in the field might have little technical know-how, and give the JOC and rangers a wide berth, the syndicates funding them may be able to provide the skills, training, and equipment necessary for someone to gain access to the anti-poaching systems and communications of an area or park.

1.5. Limitations

It is anticipated that some information regarding anti-poaching procedures, especially methods, equipment, and locations, may be difficult to find or are obscured. This is speculated to be due to the sensitive nature of the subject matter and the preference to keep such data away from public scrutiny. The consultation of a variety of sources can aid in forming a clear composite picture of a hypothetical and interoperable anti-poaching system.

While data could be sourced about anti-poaching systems and related technologies, little could be found on the specific measures applied to secure such systems. This indicates that there may be a gap in the literature, or that such data is kept private, due to the delicate nature of poaching information.

1.6. Research Questions

This thesis is based on two main research questions, alongside an additional subsidiary question. The main questions focused on here is that of:

RQ1: Can cybersecurity risks be adequately identified within a hypothetical anti-poaching system and mitigated using simulation tools?

RQ2: Can the risks identified in an anti-poaching system be calculated with threat models?

And a subsidiary question:

SQ1: What cybersecurity countermeasures are currently in place to protect the anti-poaching industry and what additional countermeasures can be added or modified to improve the safety of animals?

1.7. Brief Chapter Overviews

Chapter 1, Introduction, introduces the background for this research, and spells out the problem Statement and Research Questions, which motivates the purposes and objectives of this research.

Chapter 2, Literature Review, lays out all the relevant literature and knowledge gained from academic sources to contextualise the findings of this research. The literature will be presented in a broad overview as well as in a more detailed format for the information deemed essential.

Chapter 3, Method, delineates the methodology used to conduct the research and produce the findings. The design of the research project and the accompanying research instruments are laid out. The resulting data is gathered and analysed, threat modelling and authentication is discussed, while acknowledging any related ethical concerns, and limitations.

Chapter 4, Findings, forms the crux of the research by presenting all the findings gained from the previous chapters' work. Any important parts of the findings will be highlighted.

Chapter 5, Conclusion, is where the main findings and every other loose strand of the research is pulled together to present a coherent overview of the research. Firstly, a summary of the findings is made. The research questions will be revisited here, followed by recommendations Lastly, a section is provided for the discussion of future work and research considerations that can be made.

Chapter 6 is dedicated to the consulted references. Appendices are attached at the end of this document.

2. Literature Review

2.1. Introduction

Academic literature on anti-poaching methods, improvements, technologies, and case studies are easy to source today. With poaching being more in the public eye and having become more drastic in recent times, many conservationists are focussed on studying the poaching phenomenon. However, literature remains scarce on what anti-poaching systems are currently in place in poaching hotspots, and therefore it is even scarcer to find what, if any, cybersecurity research has been applied to such systems. This literature review will collate the available research to familiarise the reader with fundamental concepts for this paper, starting with a broad overview before delving deeper into a variety of topics, such as technologies, organisations, threat modelling and authentication.

2.2. Broad Context

As technology is constantly evolving, the anti-poaching environment is exposed to new threats and more risks, but cybersecurity is also continually adapting to this evolving technology landscape. Therefore, if one can isolate various anti-poaching technologies, and then combine them into a holistic, but hypothetical, system guided by literature, one may be afforded a clearer picture of potential anti-poaching systems and how they are set up. Thus, an overview of researched technologies and their cybersecurity concerns can aid in detailing this picture of an anti-poaching system, and the risks that threaten it. It is also noted that social engineering is a concern in anti-poaching, as it is a socio-technical system.

Drawing on guidelines issued by the South African government, and initiatives established by private companies, one can place the technologies identified here in the context of an anti-poaching system.

The core goal of this study is to determine the inherent risks in an antipoaching system, and to achieve this objective, risk, as it relates to cybersecurity, needs to be understood.

General and common threats to network systems are also discussed, from which one can calculate risk.

Lastly, the topic of identity and authentication is briefly discussed. While not a threat, its mismanagement paves the way for threats to compromise a system.

2.3. Technologies

In the last few years, academic literature has proposed or investigated different technologies and systems that can improve anti-poaching operations. Some technologies identified in the literature include those shown in Table 1.

Table 1: Technologies in use by anti-poaching operations

TECHNOLOGY	REFERENCE
Wi-Fi	Connected Conservation, 2022
CCTV	DEA, 2020
Thermal Cameras	Simlai, 2015 Connected Conservation, 2022
Camera Traps	Simlai, 2015 Hossain, *et al*., 2016 Singita, 2022 DEA, 2020
Wireless Sensor Networks	Massawe, *et al*., 2017 Yayha, *et al*., 2019
Drones	Chapman & White, 2019 Mukwazvure & Magadza, 2014 Mulermo-Pazmany, *et al*., 2014 Penny, *et al*., 2013 Simlai, 2015 Singita, 2022
CMORE	CSIR, 2022 SANParks, 2022
Nomadic (Radio) Masts	Connected Conservation, 2022
Mobile Apps	CMORE, 2022 Connected Conservation, 2022
Tags	O'Donoghue & Rutz, 2016 Bridge, *et al*., 2019
Heat Sensing Planes	Mukwazvure & Magadza, 2014

These are coupled with general systems and people not explicitly mentioned, such as computers, the cloud, security personnel, rangers, servers, radios, and the Internet.

2.4. Cybersecurity Concerns

This section provides background on some of the technologies and their relevant cybersecurity concerns.

When external attackers target a system and gain entry, it is usually termed “hacking”. This study prefers to make use of the general term “attack”, and to qualify it where necessary. Common cyber-attacks include *denial of service* (DoS) and *man in the middle* (MitM) attacks. These attacks can render systems unusable or infiltrate them to access information.

The common anti-poaching technologies and their associated vulnerabilities will now be detailed. It has been decided to omit heat sensing planes from further review, as their scope and application in the anti-poaching space is still very narrow and the related sources outdated. Nomadic Masts, Wi-Fi and CMORE will be mentioned here to provide a thorough background on anti-poaching system components but will be omitted from the hypothetical network due to the simulation capabilities of the intended software programme.

2.4.1. Wi-Fi

Guo (2019) states that Wi-Fi networks are easy to implement, but difficult to secure. The author elaborates that there are several weak holes in the Wi-Fi space and steps should be taken to protect the network and information from malicious intentions. There are many attacks that can be performed on Wi-Fi, but the key attacks identified by Guo (2019) are those of Rogue Access Points, Address Resolution Protocol (ARP) attacks, MitM attacks, and Race Condition Attacks. The author names proper configuration, end-to-end encryption, and a Wireless Intrusion Detection System as ways with which to combat Wi-Fi based attacks.

2.4.2. CCTV

Svensson and Ryden (2019) state that as CCTV cameras have moved from physical to Internet connected systems, an increasing number of cyber-attacks have been reported. This is due to the nature of these systems, with the authors stating that the exploits are basic in nature and are not usually reliant on advanced attacking capabilities. Simple mistakes, such as weak passwords and the failure to update the system, can create a vulnerable backdoor into a CCTV network.

2.4.3. Thermal Camera and Camera Traps

Davis (2021) details how camera traps have been implemented around the world as wildlife monitoring tools, specifically cameras running on Raspberry Pi (Pi) devices. A Pi is described by Cellan-Jones (2011) as an entire computer system encompassing a small circuit board. As Pi's are powerful and tiny, but inexpensive, it is ideal for deployment as the computing devices upon which wildlife camera traps run (Davis, 2021).

Like any computer, Pi's have their share of vulnerabilities and Fromaget (n.d.) states that by default, Pi's lack adequate security, which is a problem when connected to a larger or external network. This inadequacy leads Martin, Kargaard and Sutherland (2019) to state that misconfiguration of Pi devices can lead to network wide malware infections. The authors cite an OWASP report stating that easily guessable, weak and default passwords are the main vulnerability experienced by a device such as the Pi.

Thermal cameras can also be accessed, intercepted, or attacked.

Outside of cybersecurity concerns, Meek, *et al*., (2018) discovered that vandalism and theft of camera traps were a global concern in the field of wildlife monitoring, and this can further impact monitoring projects and activities.

2.4.4. Drones

Siddappaji, *et al*., (2020) mention that drones are open to cybersecurity attacks due to their singular nature. Dependent on a virtual network and various embedded computing systems, drones operate in a unique network ecosystem, encompassing remotely located physical components. Yaacoub, *et al*., (2020) states that by using wireless communications, drones are at risk. Drones are susceptible to DoS, control loop, destruction, and information corruption attacks. Due to reliance on GPS, drones can also be vulnerable to GPS spoofing attacks. Based on these findings, Siddapajji, *et al*., (2020) identify cybersecurity and reliability as key concerns for drone technology and provide a risk and protection assessment scheme for drones. Further, Yaacoub, *et al*., (2020) provide a comprehensive overview of cybersecurity vulnerabilities and exploits experienced by drones, along with mitigations.

2.4.5. Wireless Sensor Networks

A wireless sensor network is a network of components, known as nodes, that can “sense” something specific, such as sound or vibration (Massawe, *et al*., 2017). Sensors can collect data from their surroundings without necessarily encountering objects (Eloff & Lemieux, 2014). The data collected by the sensor is then transmitted to a control centre and stored on a server. Wireless sensor networks have become a useful tool for environmental monitoring (Massawe, *et al*., 2017), and can have great potential benefit for anti-poaching.

Due to their ability to collect and transmit data, sensors can be vulnerable to exploitation. If an attacker were to gain access to the greater network of devices in a reserve, they would be able to intercept the sensors’ data feeds and can therefore track a vulnerable animal. Bhushan and Sahoo (2017) have identified the most conspicuous attacks regarding wireless sensor networks. These are DoS attacks, jamming, wormhole, selective forwarding and sinkhole attacks, and the Sybil attack. Countermeasures are provided by the authors. Radhappa, *et al*., (2018) investigated vulnerabilities in wireless sensor networks and classified threats into physical, link, network and routing, and transport layer attacks.

2.4.6. CMORE

CMORE is a “cloud-based platform with both mobile and web-based applications which are used to view and contribute information to the platform” (CSIR, 2022). CMORE can be used by various parties to collaborate and coordinate from different locations, consolidating information in one organised place. The CSIR (2022) states that CMORE is a secure and private platform that has proven particularly useful in the space of anti-poaching operations.

According to Morrow (2018), cloud platforms have much of the same risks as traditional data centres, such as unauthorized access, software exploits and weak security practices. However, with cloud platforms risk responsibility management is now shared between the cloud provider and the user.

2.4.7. Nomadic Masts

The nomadic masts present in an anti-poaching system form a wireless mesh across the protected area. Such a mesh consists of fixed access points that provide network connection to clients via radio transmission (Nicholas & Alderson, 2012). According to Khan, *et al*., (2008), wireless mesh networks are prone to certain security attacks. Most listed active attacks are DoS, jamming and flooding, while eavesdropping, war driving, and traffic analysis are listed among the possible passive attacks.

2.4.8. Mobile Apps

Android and iOS are the most widely used mobile platforms. Garg and Baliyan (2021) found in their comparative review that Android was more susceptible to security concerns and breaches than iOS. It is however unknown which operating system is used by anti-poaching teams. Mutchler, *et al*., (2015) found that security vulnerabilities proliferated most freely available web apps, even affecting large and popular apps. The use of mobile web apps (apps with embedded browsers) in anti-poaching would therefore imply substantial additional risk.

2.4.9. Tags

Biologgers (a small electronic tag), and Radio Frequency Identification (RFID) tags, can be affixed to an animal or human to monitor their location and/or behaviour. It provides anti-poaching operations with the benefit of keeping a close eye on vulnerable animals and the ability to manage and coordinate patrol units (O'Donoghue & Rutz, 2016).

In the case of RFID tags, Nautiyal, *et al*., (2018) list virtual attacks such as DoS, jamming and wireless communication attacks, and physical attacks such as tag abuse and cloning as cybersecurity concerns.

A new technology in the anti-poaching arena, VHF (Very High Frequency) radio telemetry ear tags for rhinos, is currently being tested in South Africa. According to the South African Wildlife College's (SAWC) CEO, Theresa Sowry, this technology can provide better information on rhino movements and behaviour, and aid in more comprehensive monitoring (SAWC, 2022). The type of data link layer used for the VHF technology will determine what vulnerabilities are present. It has been found that in an aircraft related link layer, certain data link layers are susceptible to message monitoring and hijacking, as well as DoS attacks (Yue, 2015).

2.4.10. Social Engineering

Social engineering is the act of tricking or manipulating users or organisations into divulging confidential information or performing actions that advantage the attacker. Phishing, bribing, and blackmailing are examples of social engineering activities (Salahdine & Kaabouch, 2019). With regards to network security, Salahdine and Kaabouch (2019) state that social engineering is one of the greatest dilemmas faced by it. As communication, technology, and networking increases, so does the prevalence and sophistication of social engineering attacks.

Citing the U.S. Department of Justice, Salahdine and Kaabouch (2019) report that "social engineering attacks are one of the most dangerous threats in the world." The authors found that recent surveys and studies indicate that social engineers commit 84% of cyber-attacks, and that they experience a good success rate. Humans are evidently extremely vulnerable to social engineering attacks.

Salahdine and Kaabouch (2019) believe that humans' natural tendency to trust other humans, and their inclination to favour people over technology, make the cybersecurity chain very vulnerable to social engineering attacks. This is because no matter how extensive and impenetrable a system or network has been made, if one person shares sensitive data, an attacker may be able to gain entry and bypass all security measures. Attackers typically leverage social engineering if they find there is no other way the system can be attacked, says Salahdine and Kaabouch (2019). Social engineering attacks can only be prevented by creating awareness and providing frequent training on the subject (Burrows, 2022).

In a South African-based survey, Anna Collard (a content strategist) found that half of the respondents have been exposed to an increasing number of social engineering attacks in the last year. A rising number of reports indicate that chat applications and cell phones were the main platforms that users were being targeted on (Burrows, 2022).

In a poll run across several African countries, 33% of respondents were compromised by social engineering, and in South Africa alone, 34% were victims of phishing. It was discovered that more than half of these people fell victim to phishing links while they were preoccupied. Collard states that 15-20% of employees will be negligent with regards to security. This paves the way for a significant number of the workforce to be compromised and be susceptible to social engineering attacks (Burrows, 2022).

Social engineering is mentioned here to show that it has been considered as a risk, and to acknowledge how prevalent it is, specifically in South Africa, but the scope of this study does not allow for the incorporation of an in-depth social engineering analysis into the overarching narrative.

2.5. Governmental Guidelines

The then South African Department of Environmental Affairs (DEA) (now known as the Department of Forests, Fisheries, and the Environment (DFFE)) formulated a guideline regarding anti-poaching systems (Department of Environmental Affairs, 2020) – “Guideline to Inform Decisions on the Establishment of Anti-Poaching Related Systems and Services”. In this document the DEA provides a threat management framework and suggests some of the technologies that can be used when implementing an anti-poaching capability in a protected area. This guideline includes some of the following:

- schematics for electronic gate access controls
- recommendations for sensing, detecting, and tracking suspected poaching activities
- the establishment of a JOC
    - Where personnel are appointed that “manages the integrity of the data within the sense making software platform”
- the use of Internet, radios, satellite phones and GPS for communication

This document therefore provides further data points for consideration for an anti-poaching system. Another governmental document to consider is the DEA’s “National Strategy for The Safety and Security of Rhinoceros Populations in South Africa” (Department of Environmental Affairs, 2010). This document has a strategy section (strategy number four) that refers to the establishment of an “Integrated and coordinated national information management system for all information related to rhino species in order to adequately inform security related decisions” At this stage, it is assumed that CMORE fulfils this role.

As part of strategy four’s outcomes, the DEA (2010) lists regular and extensive risk assessments as an important activity for this strategy.

2.6. Private Companies

Dimension Data and Cisco established the Connected Conservation Foundation (CCF) in 2015, a technology-backed anti-poaching initiative (BusinessWire, 2018 & Connected Conservation, 2022). Using a multitude of interoperable technologies, as seen in Appendix A, Figure 45, the CCF has succeeded in drastically curbing poaching numbers in a South African reserve (Dimension Data, 2020). However, with this blanket of technology surrounding a reserve, criminals may still attempt to exploit holes in the network's security to gain access to the reserve and the wildlife.

2.7. Risk

"Risk" has been defined in numerous ways. Peltier (2010) defines risk as "The combination of threat, probability, and impact expressed as a value in a pre-defined range". Others have tried to boil risk down to a logical equation. Kaplan and Garrick (1981) move from risk being the composite of uncertainty and damage, to risk being the division of hazards by safeguards. In the end, the authors express risk as:

$$Risk = probability + consequence$$

Equation 1: Risk equation by Kaplan and Garrick (1981)

Kaplan and Garrick (1981) were however speaking of risk in general terms. Sotnikov's (2022) equation:

$$Risk = Threat\ x\ Vulnerability\ x\ Asset$$

Equation 2: Risk equation by Sotnikov (2022)

Equation 2 is more applicable to the topic of technical cybersecurity risks. In this equation, threat refers to the probability of a threat occurring, vulnerability refers to the likelihood of a threat succeeding to exploit a weakness and asset refers to the criticality rating of the asset.

A precise definition of risk may not be necessary for the problems dealt with in this study but understanding that risk involves a potential event or threat, and its undesired effects, is sufficient.

Risk can appear in different forms and therefore different categories of risk exist. Ciampa (2015) lists the various types of risk, such as strategic or environmental risks, but the scope of this study will solely concern itself with technical risks. Technical risks are risks related to information systems, and these risks can also be called cybersecurity risks.

Threat modelling is a methodology where anticipated risks, or threats, are modelled, to aid in determining effective countermeasures (Gonzales, 2022). This study will focus on the expansion and development of several threat models, to see if threat models can aid in determining risk.

2.8. Threats

According to Borges (2021), the most prevalent threats faced by any information system are:

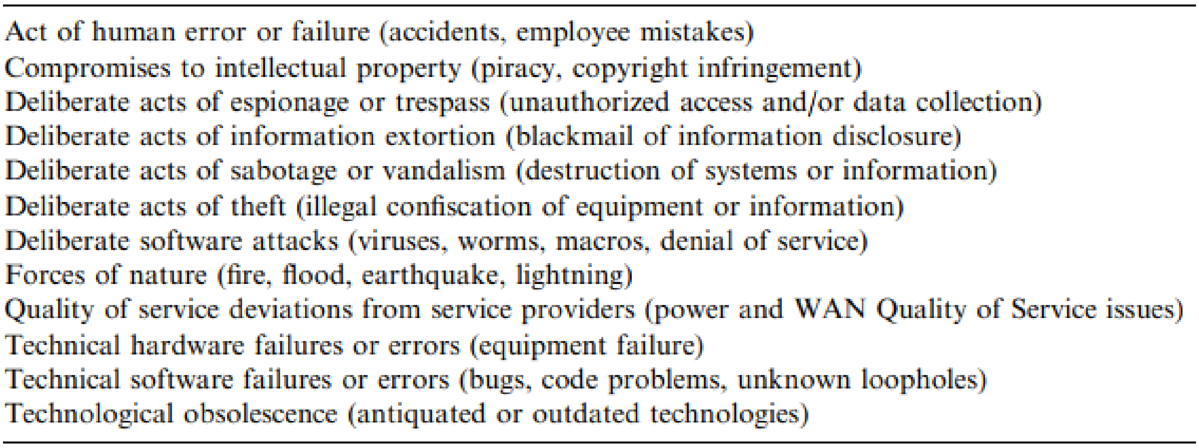
1. Malware (Microsoft, n.d.)
    - i. Computer Viruses
    - ii. Rogue Security Software
    - iii. Trojans
    - iv. Spyware
    - v. Worms
2. DoS Attacks
3. Phishing
4. Rootkits
5. SQL Injections
6. MitM Attacks

An element of this study is to see if an anti-poaching system can be virtually attacked, but these are not the only technical threats to such a system. The threats listed above will not be explained in detail but will later be considered when formulating a threat model.

Naagas and Palaoag (2018) investigated the network security of a campus information system and identified a comprehensive list of threats, which they tabled (Appendix B, Table 18), along with countermeasures. Many of these threats are inherent in any network and thus the authors' threats can be applied to an anti-poaching information system. The authors' findings will be incorporated into the threat modelling process.

Whitman (2004) performed a study to identify threats to information systems. Whitman identified various general groupings of threats, and some of these threats are useful to consider here as technical threats, but some threats, such as “Forces of nature”, fall outside of this study’s technical scope. Table 2 shows the threats identified by Whitman (2004).

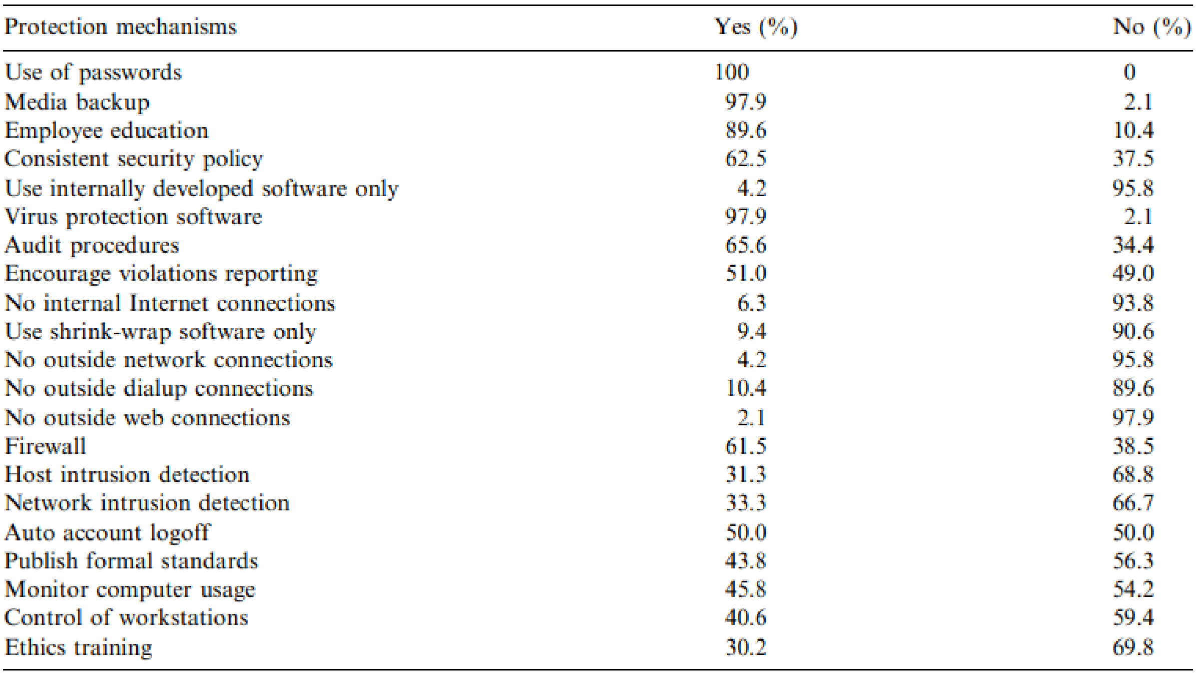
Table 2: Information system threats identified by Whitman (2004)



Act of human error or failure (accidents, employee mistakes)
Compromises to intellectual property (piracy, copyright infringement)
Deliberate acts of espionage or trespass (unauthorized access and/or data collection)
Deliberate acts of information extortion (blackmail of information disclosure)
Deliberate acts of sabotage or vandalism (destruction of systems or information)
Deliberate acts of theft (illegal confiscation of equipment or information)
Deliberate software attacks (viruses, worms, macros, denial of service)
Forces of nature (fire, flood, earthquake, lightning)
Quality of service deviations from service providers (power and WAN Quality of Service issues)
Technical hardware failures or errors (equipment failure)
Technical software failures or errors (bugs, code problems, unknown loopholes)
Technological obsolescence (antiquated or outdated technologies)

Whitman (2004) further performed surveys to observe to what extent and with what methods these threats were mitigated in practice by businesses, and their findings can be seen in Table 3. The “Yes” and “No” fields indicate the percentage of respondents that either do or do not make use of the specified protection mechanism. Table 3 provides further measures to consider for the security of information systems.

Table 3: Threat protection mechanisms identified by Whitman (2004)



Protection mechanisms	Yes (%)	No (%)
Use of passwords	100	0
Media backup	97.9	2.1
Employee education	89.6	10.4
Consistent security policy	62.5	37.5
Use internally developed software only	4.2	95.8
Virus protection software	97.9	2.1
Audit procedures	65.6	34.4
Encourage violations reporting	51.0	49.0
No internal Internet connections	6.3	93.8
Use shrink-wrap software only	9.4	90.6
No outside network connections	4.2	95.8
No outside dialup connections	10.4	89.6
No outside web connections	2.1	97.9
Firewall	61.5	38.5
Host intrusion detection	31.3	68.8
Network intrusion detection	33.3	66.7
Auto account logoff	50.0	50.0
Publish formal standards	43.8	56.3
Monitor computer usage	45.8	54.2
Control of workstations	40.6	59.4
Ethics training	30.2	69.8

However, Whitman (2004) does not provide a deeper analysis of threats and their relevant protection mechanisms, with the identified threats never matched to the necessary protection mechanism(s). This is understandable, as there are too many threats and attack methods to cover in a single article, but this may leave the reader with an unclear understanding of how to counteract various threats. Enough information is provided to the reader to understand the types of threats to an information system, and from there one can undertake further research to understand which specific threats apply to one’s own information system. From there practical and applicable countermeasures can be identified.

From Table 3, it can be seen that passwords were used by all respondents. Identity and multi-factor authentication (MFA) are an important requirement for a secure anti-poaching system and is in general widely employed in information systems.

2.8.1. Threat Models

To formulate a pre-emptive understanding of cybersecurity threats, one can approach the problem with threat modelling. During threat modelling, threats are isolated, and practices are developed to find and counter threats. To do this, one requires an understanding of how the relevant threats may affect one's information systems and how they can be categorised (Gonzalez, 2022).

Per Gonzalez's (2022) list, threat modelling generally consists of five phases. These are:

1. Gathering threat data
2. Identifying assets
3. Formulating mitigations
4. Assessing risk
5. Outlining threats

Gonzalez (2022) describes these steps as vital to an organisation's security posture. A security posture, according to Rosencrance (2022), denotes how robust an organisation's cybersecurity is overall, measured by its prediction, prevention, and response capabilities to evolving cybersecurity threats. A security posture gains strength from the benefits of threat modelling, where threats are prioritised, defences are demarcated, new tools are introduced and security gaps are fixed (Gonzalez, 2022).

There are many threat models considered by Gonzalez (2022), among them a popular model such as STRIDE. Another threat model, DREAD (Pevnev, *et al.*, 2021), is also considered. It is not within this study's scope to expand on all the threat models, but the chosen models will be detailed.

Of the threat models that were investigated, it appeared that Attack Trees, STRIDE and DREAD could prove the most useful for this research's case study. Threat models are not always used in isolation and combining multiple models can provide a more holistic picture of the threat landscape.

It was decided that Attack Trees, or tree diagrams in general, will be used to showcase overviews of the existing threats and threat scenarios. Additionally, for the most prevalent network threats, STRIDE will be used to classify threats and then the threat's risk will be estimated using DREAD.

2.9. Authentication

Strong authentication mechanisms make it more difficult for an attacker to breach a system. Authentication can be seen as the first line of defence for a system, with its failure causing a major risk to the system.

As authentication mechanisms have become a necessity in today's interconnected landscape, passwords are seen by Almehmadi and Alsolami (2019) as a "basic element" of cybersecurity strategies. Passwords are important as more data becomes digitised, but can potentially be disclosed during a cyber-attack. Almehmadi and Alsolami (2019) state passwords to be "the most feasible method of authenticating the access of individuals", but caution that with progressively sophisticated cyber-attacks, the principal use of passwords was proven to be largely ineffective when faced with an attack.

While ubiquitous, passwords inherently need to strike a compromise between usability and security. "Strong" passwords are more difficult for an attacker to crack, while at the same time it may be harder for the user to remember (Almehmadi & Alsolami, 2019). Yıldırım and Mackie (2019) concur, stating that the security risks with passwords seldom arise from technical problems, and rather stem from the limits of people's memories. These limitations may lead to insecure practices such as the user writing down passwords, opting for weaker (more guessable) passwords, using personal information, reusing passwords, and using generic or formulaic passwords (Almehmadi & Alsolami, 2019).

Insecure password practices in organisations typically result from organisational policies mandating a user to frequently change their passwords (often to complex character strings), and studies point to a general annoyance with such policies by employees (Almehmadi & Alsolami, 2019).

Almehmadi and Alsolami (2019) conclude that implementing two-factor authentication (2FA), or MFA, may be an effective way to deal with password vulnerabilities. This means an extra step(s) is added to the authentication activity and can be implemented in the form of biometrics or PINs. (The preferred term for this study will be MFA, as it not always the case that authentication is limited to only two factors). Ion, Reeder and Consolvo (2015) have found that most cybersecurity experts do make use of MFA, while non-experts preferred to use regular password changes. Ometov, *et al*., (2019) and Ali, Dida, and Sam (2020) provide an overview of different MFA methods.

Another solution to password security issues are password managers (CERN, 2018), which Ion, Reeder and Consolvo (2015) found implemented by experts in the field, while non-experts remained wary of them. In Ion, Reeder and Consolvo's (2015) conclusion, they prescribe three pieces of security advice:

1. Ensure software is up to date
2. Employ a password manager
3. Use MFA for online accounts

The above list underscores how prominent passwords are in the security chain, as they remain the point of entry to computer networks and information systems.

2.10. Conclusion

There are numerous cybersecurity concerns in the various technologies that are integrated into anti-poaching systems, and a detailed expansion on these concerns and countermeasures is not within the scope of this study. The literature review was intended solely to inform about technologies and concerns, and not to solve each of them.

It can be gleaned from the technologies, governmental guidelines and private companies' initiatives that simulating or modelling a hypothetical anti-poaching system is feasible.

Further, as the concept of risk has been explained, risk calculations and threat modelling will be presented and expanded on.

As authentication is also a risk to anti-poaching systems, various mechanisms can now be discussed and solutions considered.

3. Method

3.1. Introduction

As the reader has gained an understanding of the background for the research done, the research can be presented by first describing the research design and listing the research tools and instruments, along with applicable motivations.

The main data will then be presented in the form of the simulated hypothetical anti-poaching system, together with descriptions for the various components and sections. After the network configuration is explained, the chosen attacks will be performed and mitigated. Authentication methods will also be investigated.

In the case of limitations, they will be detailed, and ethical considerations will be noted.

3.2. Research Design

Overall, this research will be captured in a theoretical cyber-risk framework. Afribary (2021) describes a theoretical framework as one where a "general representation of relationships between things in a given phenomenon" is provided. This study aims to represent the status of cybersecurity in anti-poaching systems and recommend solutions.

The technologies or systems used in anti-poaching operations, and their resultant cybersecurity considerations, will be investigated. A mostly qualitative approach (with quantitative elements for the risk calculations) will be used to conduct research on what technology is in place, what cybersecurity measures are necessary and how the cybersecurity of the system can be improved.

A qualitative approach is useful for the contextual understanding of real-life phenomena, according to Recker (2013). It is also said to be a useful approach for exploratory research. Therefore, a qualitative methodology will suit this research, as it aims to explore cybersecurity in the context of anti-poaching systems.

Rot (2008) provides the benefits and drawbacks of qualitative versus quantitative approaches, especially in the field of IT risk management. While quantitative approaches may provide a more accurate reflection of risk, qualitative approaches are easier and quicker to use, while having the ability to prioritise risks. Qualitative approaches' downsides include gaining only a general approximation of risk, and difficulty with ascribing costs to risks. However, these disadvantages do not impact the research here, as costs will not be calculated and proving the general presence and degree of cyber risk is deemed sufficient.

After it has been determined what compromises a hypothetical anti-poaching system, this system can be attacked. Simulating and performing attacks can provide an indication if such attacks pose a real risk to the system. With data gained on these attacks and their mitigations, in addition to other already present threats named in the literature, the combined knowledge can be captured via threat modelling. The aim here is to provide different views of threats in a system, while attempting to provide a quantifiable way of estimating risk in the system.

This thesis does not intend to completely overhaul or replace existing cybersecurity structures and perform comparative evaluations, nor does it intend to perform real physical penetration testing on existing systems.

This thesis further aims to ground itself in the South African context, as it is rich in wildlife.

> “South Africa is now the last country in the world with a significant population of rhinos left in the wild. This is one of the reasons why it is bearing the brunt of what can arguably be described as one of the worst global wildlife conservation crises of the past 100 years. (ZAPWing, 2022)

As the above ZAPWing (2022) quote indicates, South Africa (specifically the Kruger National Park), is home to the majority of the globe’s rhinos (Save the Rhino, 2020). Rhinos are the key poaching casualties, and therefore South Africa is a particularly useful country to focus on, since there will be many anti-poaching operations taking place throughout the country.

This thesis will not exclusively concentrate on rhino related literature, but expand its focus to all wildlife, if possible. This will be done as anti-poaching systems in parks and reserves are not solely focused on rhinos, but species such as elephants, lion, pangolin, and others are also in need of monitoring and protection. However, it is acknowledged that the literature does still place an emphasis on rhino and elephant poaching.

3.3. Research Instruments

The main software programmes used to conduct the research were GNS3 and VMWare, in addition to the Kali Linux operating system, which is home to a suite of network attack tools.

The specifications of the machine powering the software is as follows:

- 11th Gen Intel® Core™ i7-11800H @ 2.30GHz (16 CPUs)
- 16 GB DDR4 RAM
- Windows 10 OS
- NVIDIA GeForce RTX 3060 6 GB Laptop GPU

3.3.1. Graphical Network Simulator 3 (GNS3)

Graphical Network Simulator, or GNS3 (GNS3, 2022a), is a freely available and open-source network simulator where one can build network topologies based on real life systems and components. It mostly features Cisco components, but also runs Juniper, MikroTik and Arista hardware, and can import images of various operating systems (Neumann, 2015).

GNS3 is the overlay for other open-source programmes, such as Dynamips (a Cisco emulator), QEMU (Quick Emulator), and VirtualBox (which can emulate Windows and Linux OS's) (Welsh, 2013).

Commenting on GNS3, Neumann (2015) names the customisation abilities, flexibility, low cost, scalability, portability and virtual to physical bridging as key advantages of the GNS3 software. It offers the ability to access projects at any point, as an Internet connection is not necessary. Theoretically an unlimited number of projects can be created with an unlimited number of objects or components.

Practically, the more objects or components added to a project, the more likely it is that the programme will start to exceed the available memory and processing power of the host machine (Welsh, 2013 & Neumann, 2015). Another downside of GNS3 is that of a Dynamips limitation – the programme is beneficial and sufficient for educational purposes but does not allow for deployment in a larger production environment. (Neumann, 2015)

For optimal functionality, GNS3 recommends working in a Virtual Machine (VM) environment while building one's topology. Therefore, the installation of an accompanying programme is required, such as that of VMWare (GNS3, 2022b). GNS3 works in tandem with a tool called SolarPuTTy, which allows one to access the consoles of the network components and run commands.

GNS3 (version 2.2.31) was chosen here specifically for its free availability, suitability to the proposed network topology and its ease of use. Further, the host machine used was verified to meet the recommended technical specifications of GNS3 (Welsh, 2013 & GNS3, 2022c).

3.3.2. VMWare

While offering a paid, Workstation Pro version of their programme, VMWare has created VMWare Workstation Player, a freely available VM programme for non-commercial use. According to VMWare (2022) one can "easily run multiple operating systems as virtual machines on your […] PC". Therefore, VMWare Workstation Player (version 16) appeared sufficient for the needs of the research project, in addition to being GNS3's VM programme of choice.

3.3.3. Kali Linux

Cesar and Pinter (2019) state that "Kali Linux is a Debian-based Linux distribution focused on advanced penetration testing and ethical hacking" (from here on Kali Linux will be simply referred to as Kali). Developed by Offensive Security, Kali is very popular in the cybersecurity industry and features a pantheon of penetration testing, ethical hacking and computer forensic tools and utilities (Cesar & Pinter, 2019).

Kali (distribution 2021.1), as an attacking machine, was chosen for its ease of use, great performance in a VM environment, its numerous well-documented cybersecurity-related tools, and for its suitability for educational use (Cesar & Pinter, 2019).

The following attacking tools were utilised:

- Yersinia: Yersinia comes packaged with Kali and provides tools to perform what is known as Layer 2 attacks. Yersinia can be run from the terminal or via a GUI, as was the case here (Kali, 2022).
- Ettercap: The pre-installed Ettercap tool features many MitM attack utilities, and its GUI was found easy to navigate (Ettercap, 2022).
- hping3: hping3 was installed and used for its ability to send custom TCP packets over a network (Kali, 2021a).

- macof: Part of the Dsniff suite of tools that can be installed on Kali, macof provides one with the capability to flood a switch with MAC addresses (Sankar, 2015)
- Nmap: Nmap is a well-known open-source network discovery and mapping tool that is included in the Kali suite (Nmap, 2022).
- Wireshark: “the world’s foremost … protocol analyzer” is how Wireshark describes itself. Freely available, Wireshark is used in both Kali and GNS3 (Wireshark, 2022).

3.4. Hypothetical Anti-Poaching System Network

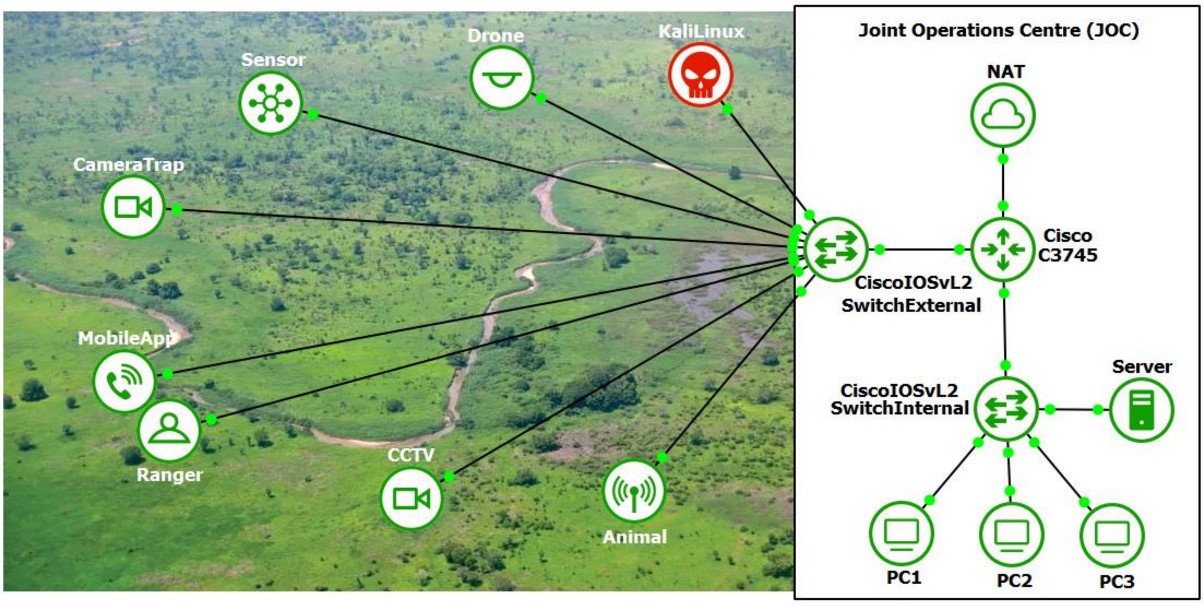


Figure 1: Hypothetical anti-poaching network

Figure 1 depicts the entirety of the hypothetical network topology. A JOC has been set up, from where all devices are connected and from where all monitoring of field devices and sensors can take place. The design and implementation of the hypothetical network topology was partially adapted from the CCF diagram in Appendix A (Figure 45), and the DEA’s guideline document.

This design represents the main network which will be attacked and where it will be attempted to mitigate these attacks. Most attacks will focus attacking SwitchExternal, which will in turn influence the other network components’ communication abilities.

Table 4 provides a description of each component featured in Figure 1’s network design.

Table 4: Description of anti-poaching network components

Component	Symbol	Topology Example(s)	Description
NAT Cloud	☁	NAT	Provides Internet connectivity to network components via the router (GNS3, 2022d)
Router	✥	Cisco C3745	Connects two or more networks and facilitates traffic between them by enabling devices to use the same Internet connection and sending packets to the correct IP addresses (Cloudflare, 2022)
Switch	⇄	Cisco IOSvL2 SwitchExternal, Cisco IOSvL2 SwitchInternal	Channels access to the Internet and components via the router to the various components connected to it. Capable of connecting to more components than the router in GNS3, it is therefore a necessity for larger networks (Shaw, 2020)
Virtual PC	🖥	PC1-3	Personal Computers (PCs) used to monitor incoming signals and access cloud and web platforms
	⊖	Drone	Drone, or Unmanned Aerial Vehicle (UAV), used to monitor an area by being able to fly overhead and transmit video and sound (Mulero-Pazmany, *et al*., 2014)
	✲	Sensor	Sensors can detect various signals and phenomena, such as gunshots, voices, and movement (Massawe, *et al*., 2017)
	☎	MobileApp	Mobile app is used to access web and cloud platforms and communicate with the JOC (Connected Conservation, 2022 & CSIR, 2022)
	🎥	CameraTrap	Camera traps capture images based on movement in front of the camera (Simlai, 2015)
	🎥	CCTV	CCTV monitors wide areas with a camera feed from high vantage points, such as up on poles or below a roof (Connected Conservation, 2022)
	👤	Ranger	Tracking tag on rangers in anti-poaching units to monitor their movements and location (Connected Conservation, 2022)
	📶	Animal	Tracking tags on key animals in in the field to monitor their movements and location (O’Donoghue & Rutz, 2016)
Server	🖥	Server	Server acts as a database and storage location for the information gathered and kept in the JOC (Ingalls, 2021). In the topology, Windows Server 2016 is the operating system
Virtual OS	☠	KaliLinux	Intruder device running the Kali Linux operating system. Used by attacker to try and gain access to the network (Bergs, et al., 2021)

3.4.1. Joint Operations Centre (JOC)

A closer and isolated picture of the JOC can be seen in Figure 2.

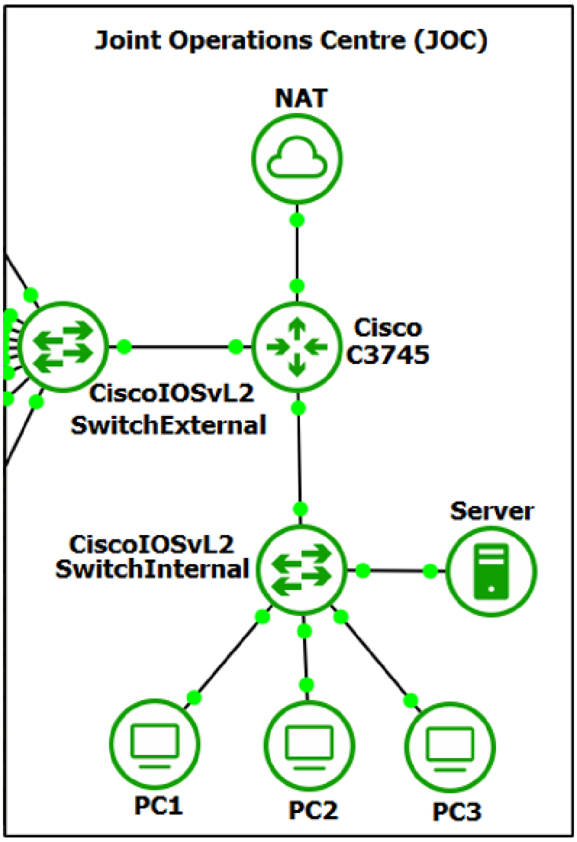


Figure 2: JOC of the anti-poaching network

While the full hypothetical network topology can only be estimated to a certain extent, there exists more concrete guidelines for the setting up of a JOC. These guidelines are contained in the “Guidelines to Inform the Establishment of Anti-Poaching Related Systems and Services” paper issued by the then South African DEA (Department of Environmental Affairs, 2020).

According to the guideline, the basic functions of the JOC can be achieved with only a laptop, cell phone and one dedicated person. Ideally, the JOC will be staffed by at least three staff members, with a workstation each and three extra workstations. The guideline specifies that at least one desktop should be dedicated to CMORE, while the others may be utilised for staff and administration work. It was decided that the JOC in the network topology would have only three Personal Computers (PC1-3). This is to represent the three dedicated staff members, and their functions, and it was deemed that more computers were not necessary and that they would become redundant.

The guideline proposes the establishment of a radio room, where two staff members will always be on duty. It has been decided to omit such a room, and its inherent components, from the topology. This was done as the simulation software does not provide for radio frequency modelling, and it was deemed to be unnecessary for the overall purposes of the study.

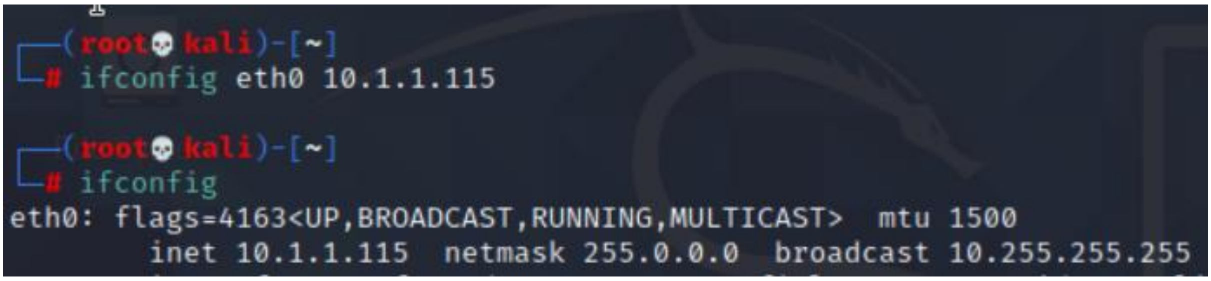
The guideline groups the JOC and radio room together with a Main Server Room. It was decided not to model a separate server room adjacent to the JOC, as this streamlines the topology and the topology’s existing server encapsulates the data storage capabilities that the JOC would have access to. Additionally, physically incorporating a server into the JOC adds an extra layer of security to the dedicated, and mostly sensitive, data generated and received by the JOC.

Only general devices are specified for the JOC. In this simulation, specific network devices, such as the router and switch, needed to be chosen. The Cisco C3745 model router was used here as it provided the necessary capabilities and was successfully utilised by Bergs, *et al*., (2021), who performed similar attacks in a similar GNS3 network topology. The Cisco IOSvL2 model switch was chosen here, for both SwitchInternal and SwitchExternal. The IOSvL2, per Cisco’s specifications (Cisco, 2022), supports a wide variety of features, that are required here to successfully perform the cyber-attacks and mitigations on the network.

3.4.2. Network Configuration

After designing the network layout, the components needed to be correctly configured to communicate with the Internet and each other before any attacks could be performed.

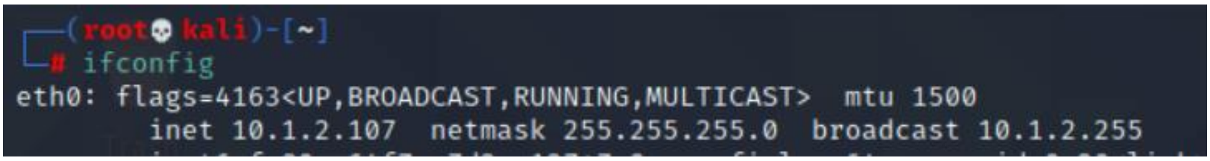
Firstly, the C3745 router was connected to the Internet via the NAT Cloud. The C3745 router was then configured as a DHCP (Dynamic Host Configuration Protocol) server, as it was the Internet access point for all components of the system. DHCP, with a gateway of 10.1.1.1, was then enabled on the C3745 router. The IP pool for SwitchInternal was allocated from 10.1.1.100 upwards. The IP pool for SwitchExternal started at 10.1.2.100. The IP of 10.1.1.115 was added as an excluded address on the DHCP server, as this IP was reserved as a static IP for the KaliLinux component. Figure 3 shows how Kali’s IP was manually reconfigured to 10.1.1.115, using the “ifconfig eth0 10.1.1.115” command.



```
┌──(root💀kali)-[~]
└─# ifconfig eth0 10.1.1.115

┌──(root💀kali)-[~]
└─# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.1.1.115  netmask 255.0.0.0  broadcast 10.255.255.255
```

Figure 3: Kali being assigned a manual IP address

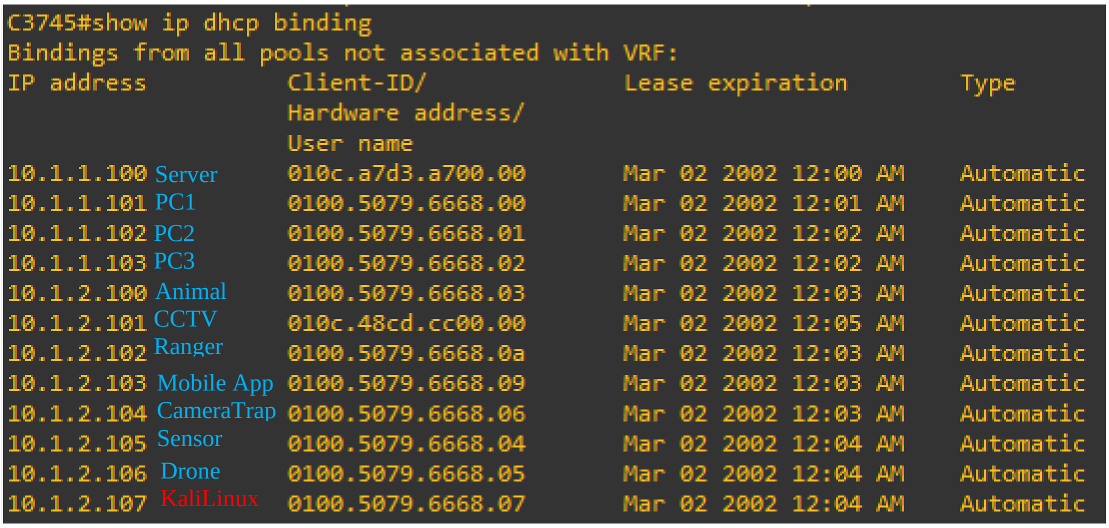
However, after being assigned this IP the Kali machine would not connect to the Internet or network and it was decided that it would be ensured that it always has the same DHCP address (10.1.2.107), shown in Figure 4. This was done so that any malicious activities originating from this component can be tracked and monitored if needed.



```
┌──(root💀kali)-[~]
└─# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.1.2.107  netmask 255.255.255.0  broadcast 10.1.2.255
```

Figure 4: Kali's DHCP assigned IP address

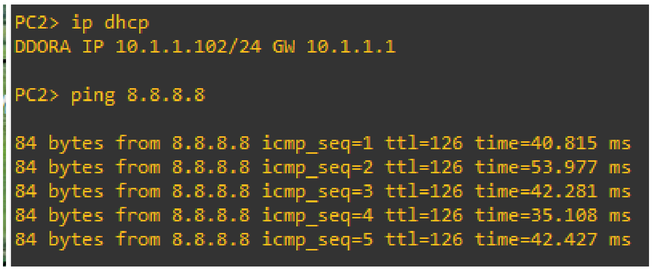
All the other components in the networks then received IPs from the DHCP server, as can be seen in Figure 5, after running the command "show ip dhcp binding" in the console of the C3745 router.



```
C3745#show ip dhcp binding
Bindings from all pools not associated with VRF:
IP address          Client-ID/              Lease expiration        Type
                    Hardware address/
                    User name
10.1.1.100 Server   010c.a7d3.a700.00       Mar 02 2002 12:00 AM    Automatic
10.1.1.101 PC1      0100.5079.6668.00       Mar 02 2002 12:01 AM    Automatic
10.1.1.102 PC2      0100.5079.6668.01       Mar 02 2002 12:02 AM    Automatic
10.1.1.103 PC3      0100.5079.6668.02       Mar 02 2002 12:02 AM    Automatic
10.1.2.100 Animal   0100.5079.6668.03       Mar 02 2002 12:03 AM    Automatic
10.1.2.101 CCTV     010c.48cd.cc00.00       Mar 02 2002 12:05 AM    Automatic
10.1.2.102 Ranger   0100.5079.6668.0a       Mar 02 2002 12:03 AM    Automatic
10.1.2.103 Mobile App 0100.5079.6668.09     Mar 02 2002 12:03 AM    Automatic
10.1.2.104 CameraTrap 0100.5079.6668.06     Mar 02 2002 12:03 AM    Automatic
10.1.2.105 Sensor   0100.5079.6668.04       Mar 02 2002 12:04 AM    Automatic
10.1.2.106 Drone    0100.5079.6668.05       Mar 02 2002 12:04 AM    Automatic
10.1.2.107 KaliLinux 0100.5079.6668.07      Mar 02 2002 12:04 AM    Automatic
```

Figure 5: Network's list of DHCP IP address (labels added)

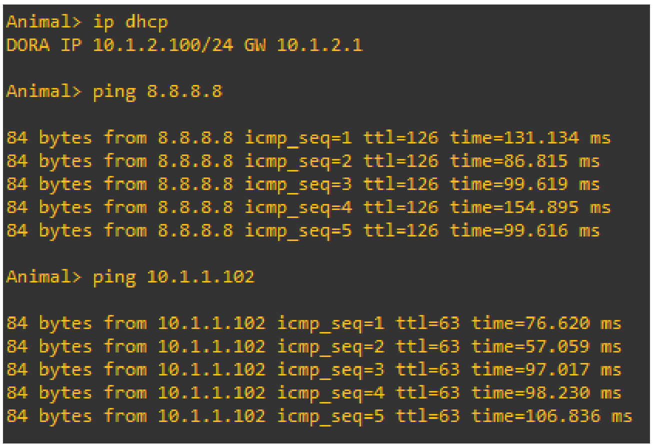
In Figure 6, one can see the PC2 component receiving the 10.1.1.102 IP address from the DHCP server, and the “ping 8.8.8.8” (a ping to Google) command shows that PC2 can now send and receive packets over the Internet, indicating a successful connection via the switch and the router.



```
PC2> ip dhcp
DDORA IP 10.1.1.102/24 GW 10.1.1.1

PC2> ping 8.8.8.8

84 bytes from 8.8.8.8 icmp_seq=1 ttl=126 time=40.815 ms
84 bytes from 8.8.8.8 icmp_seq=2 ttl=126 time=53.977 ms
84 bytes from 8.8.8.8 icmp_seq=3 ttl=126 time=42.281 ms
84 bytes from 8.8.8.8 icmp_seq=4 ttl=126 time=35.108 ms
84 bytes from 8.8.8.8 icmp_seq=5 ttl=126 time=42.427 ms
```

Figure 6: PC2 pings Google

The same commands were performed on the Animal (10.1.2.100) device and the results are shown in Figure 7:



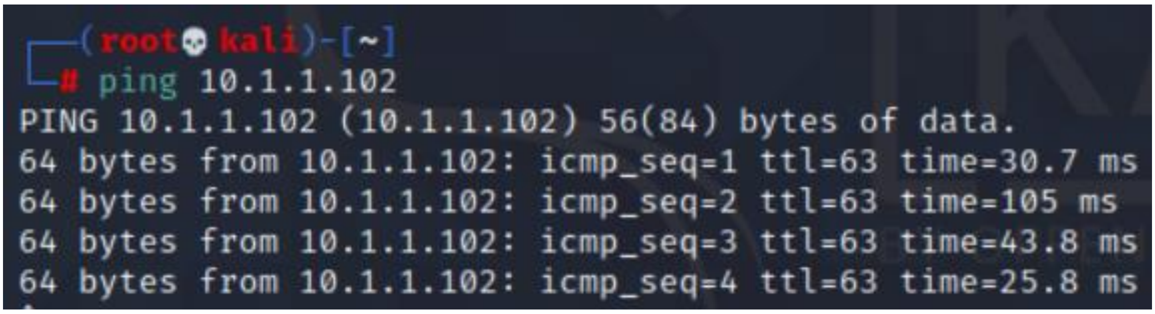
```
Animal> ip dhcp
DORA IP 10.1.2.100/24 GW 10.1.2.1

Animal> ping 8.8.8.8

84 bytes from 8.8.8.8 icmp_seq=1 ttl=126 time=131.134 ms
84 bytes from 8.8.8.8 icmp_seq=2 ttl=126 time=86.815 ms
84 bytes from 8.8.8.8 icmp_seq=3 ttl=126 time=99.619 ms
84 bytes from 8.8.8.8 icmp_seq=4 ttl=126 time=154.895 ms
84 bytes from 8.8.8.8 icmp_seq=5 ttl=126 time=99.616 ms

Animal> ping 10.1.1.102

84 bytes from 10.1.1.102 icmp_seq=1 ttl=63 time=76.620 ms
84 bytes from 10.1.1.102 icmp_seq=2 ttl=63 time=57.059 ms
84 bytes from 10.1.1.102 icmp_seq=3 ttl=63 time=97.017 ms
84 bytes from 10.1.1.102 icmp_seq=4 ttl=63 time=98.230 ms
84 bytes from 10.1.1.102 icmp_seq=5 ttl=63 time=106.836 ms
```

Figure 7: Animal successfully pings Google and PC2

Additionally, it is shown in Figure 7 that with the “ping 10.1.1.102” command, Animal can successfully communicate with PC2. Therefore, when all components have their DHCP assigned IP addresses, they can connect to the Internet and communicate with any other component in the network.

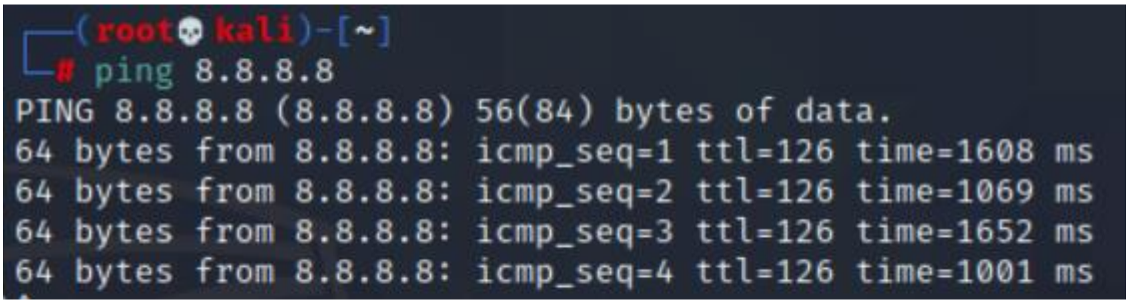
The Kali machine was also configured. Firstly, a root (a superuser or most privileged account) was activated using the “sudo apt -y install kali-root-login” command. This was done in case a tool requires special privileges later and removes the need to constantly prefix commands with “sudo” (Kali, 2021b).

The “ping 8.8.8.8” and “ping 10.1.1.102” commands were also successfully run on the Kali machine to verify that the network connections were working. This can be seen in Figures 8 and 9.



```
┌──(root💀kali)-[~]
└─# ping 10.1.1.102
PING 10.1.1.102 (10.1.1.102) 56(84) bytes of data.
64 bytes from 10.1.1.102: icmp_seq=1 ttl=63 time=30.7 ms
64 bytes from 10.1.1.102: icmp_seq=2 ttl=63 time=105 ms
64 bytes from 10.1.1.102: icmp_seq=3 ttl=63 time=43.8 ms
64 bytes from 10.1.1.102: icmp_seq=4 ttl=63 time=25.8 ms
```

Figure 8: Kali Successfully pings PC2



```
┌──(root💀kali)-[~]
└─# ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=126 time=1608 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=126 time=1069 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=126 time=1652 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=126 time=1001 ms
```

Figure 9: Kali successfully pings Google

3.5. Cyber Attacks and Mitigation

As the network has been set up and the individual components can verifiably communicate with each other, the network is now ready to be attacked. ARP poisoning, DHCP spoofing and starvation, STP attack, TCP/SYN flooding, CAM overflow, VLAN hopping, and Port scanning are the attacks that was attempted. These attacks were chosen because they are common attacks, cover different attack types, such as DoS, MitM and Layer 2 (L2) attacks, and are considered among the common threats to a network by Borges (2021). DoS attacks were also quite prominent during the discussion of the cybersecurity concerns of various anti-poaching technologies.

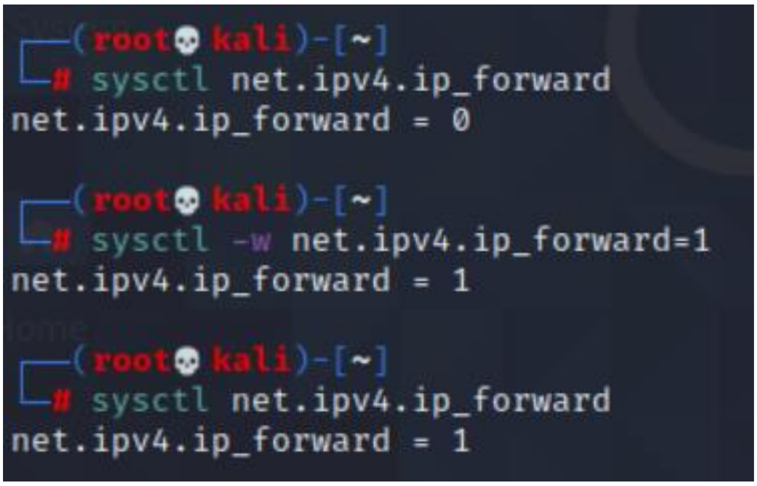
3.5.1. ARP Poisoning

In computer communications, the widely used Address Resolution Protocol (ARP) can convert IP addresses into matching MAC addresses. Due to its statelessness an unauthenticated nature, ARP presents with exploitable vulnerabilities (Tripathi & Mehtre, 2014).

ARP poisoning is deemed to be the collection of basic attacks that can predicate higher level attacks (Tripathi & Mehtre, 2014). ARP poisoning occurs when the protocol's weaknesses are exploited in attempt to jeopardise a network's security against MitM attacks, DoS attacks and session hijacking. When a host's ARP table undergoes malicious alteration by receiving falsified ARP packets, the ARP has been poisoned (Mangut, *et al.*, 2015 and Jana, 2016) Mangut, *et al.*, (2015) elaborate that ARP poisoning "is the commonest attack that can be launched from within a network and could have a very high destruction profile."

Attack:

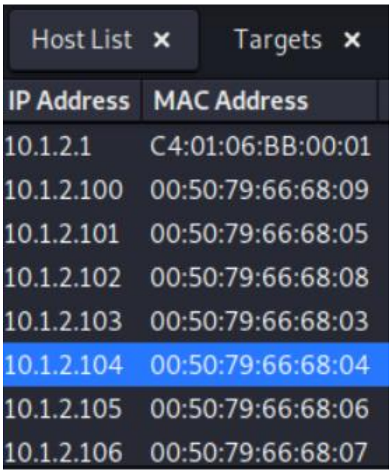
Before Ettercap was opened for the attack, IP forwarding needed to be enabled on the Kali machine, as Figure 10 shows its status is set to 0 by default, meaning it was not activated. The appropriate command was run, and one can see that IP forwarding is now enabled, indicated by the 1 value (Reynolds, 2022).



```
┌──(root💀kali)-[~]
└─# sysctl net.ipv4.ip_forward
net.ipv4.ip_forward = 0

┌──(root💀kali)-[~]
└─# sysctl -w net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1

┌──(root💀kali)-[~]
└─# sysctl net.ipv4.ip_forward
net.ipv4.ip_forward = 1
```

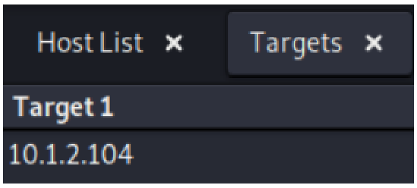
Figure 10: IP forwarding is activated and verified on Kali

Ettercap was then first instructed to sniff for viable hosts on the network to exploit. Due to the Kali machine being located on the 10.1.2.1 switch (SwitchExternal), Ettercap could only identify IPs on SwitchExternal and did not sniff IPs on SwitchInternal, or on the C3745 router. The sniffed IP address can be seen in Figure 11.



IP Address	MAC Address
10.1.2.1	C4:01:06:BB:00:01
10.1.2.100	00:50:79:66:68:09
10.1.2.101	00:50:79:66:68:05
10.1.2.102	00:50:79:66:68:08
10.1.2.103	00:50:79:66:68:03
10.1.2.104	00:50:79:66:68:04
10.1.2.105	00:50:79:66:68:06
10.1.2.106	00:50:79:66:68:07

Figure 12: Ettercap's sniffed IP adresses

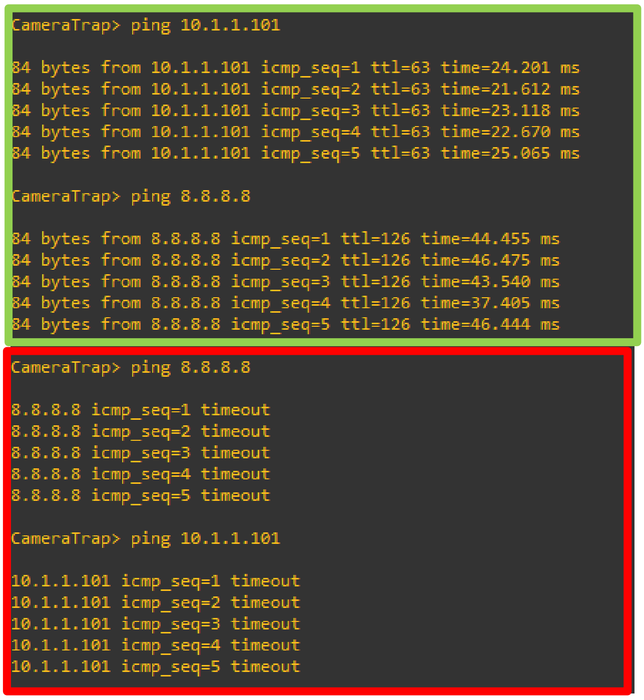


Target 1
10.1.2.104

Figure 11: CameraTrap added to target list in Ettercap

CameraTrap (10.1.2.104) was chosen as the victim to attack and was selected from the host list to be added to the target list, shown in Figure 12.

The ARP poisoning attack option was then selected within Ettercap, and the attack was launched on the target CameraTrap. Figure 13 shows the state of the target before and after the attack. The green block indicates CameraTrap’s ability to communicate with both Google and PC1 before the attack, and red showcases the ability after the attack was launched.

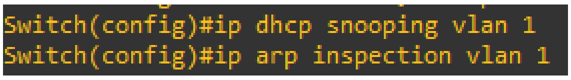


```
CameraTrap> ping 10.1.1.101

84 bytes from 10.1.1.101 icmp_seq=1 ttl=63 time=24.201 ms
84 bytes from 10.1.1.101 icmp_seq=2 ttl=63 time=21.612 ms
84 bytes from 10.1.1.101 icmp_seq=3 ttl=63 time=23.118 ms
84 bytes from 10.1.1.101 icmp_seq=4 ttl=63 time=22.670 ms
84 bytes from 10.1.1.101 icmp_seq=5 ttl=63 time=25.065 ms

CameraTrap> ping 8.8.8.8

84 bytes from 8.8.8.8 icmp_seq=1 ttl=126 time=44.455 ms
84 bytes from 8.8.8.8 icmp_seq=2 ttl=126 time=46.475 ms
84 bytes from 8.8.8.8 icmp_seq=3 ttl=126 time=43.540 ms
84 bytes from 8.8.8.8 icmp_seq=4 ttl=126 time=37.405 ms
84 bytes from 8.8.8.8 icmp_seq=5 ttl=126 time=46.444 ms
CameraTrap> ping 8.8.8.8

8.8.8.8 icmp_seq=1 timeout
8.8.8.8 icmp_seq=2 timeout
8.8.8.8 icmp_seq=3 timeout
8.8.8.8 icmp_seq=4 timeout
8.8.8.8 icmp_seq=5 timeout

CameraTrap> ping 10.1.1.101

10.1.1.101 icmp_seq=1 timeout
10.1.1.101 icmp_seq=2 timeout
10.1.1.101 icmp_seq=3 timeout
10.1.1.101 icmp_seq=4 timeout
10.1.1.101 icmp_seq=5 timeout
```

Figure 13: CameraTrap's ping ability before (green) and during ARP Poisoning (red)

Mitigation:

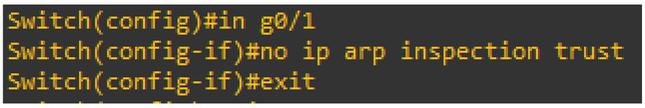
Mitigation measures were applied as specified by Mangut, *et al.*, (2015) and Bergs, *et al.*, (2020). DHCP snooping and ARP inspection were activated on VLAN 1 on SwitchExternal, as seen in Figure 14:



```
Switch(config)#ip dhcp snooping vlan 1
Switch(config)#ip arp inspection vlan 1
```

Figure 14: Mitigations applied against ARP Poisoning

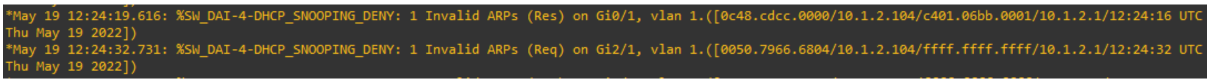
After this was applied, the Kali machine’s port on the switch, g0/1, was disallowed trust, as seen in Figure 15:



```
Switch(config)#in g0/1
Switch(config-if)#no ip arp inspection trust
Switch(config-if)#exit
```

Figure 15: Disallowing ARP trust for Kali

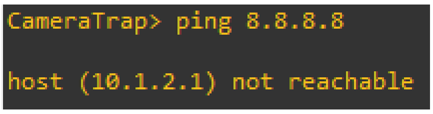
The Ettercap ARP poisoning attack was launched again. It was noticed that the switch was now providing alerts regarding DHCP snooping and ARP inspection on both Kali’s ports and that of CameraTrap’s (g2/1). This can be seen in Figure 16.



```
*May 19 12:24:19.616: %SW_DAI-4-DHCP_SNOOPING_DENY: 1 Invalid ARPs (Res) on Gi0/1, vlan 1.([0c48.cdcc.0000/10.1.2.104/c401.06bb.0001/10.1.2.1/12:24:16 UTC
Thu May 19 2022])
*May 19 12:24:32.731: %SW_DAI-4-DHCP_SNOOPING_DENY: 1 Invalid ARPs (Req) on Gi2/1, vlan 1.([0050.7966.6804/10.1.2.104/ffff.ffff.ffff/10.1.2.1/12:24:32 UTC
Thu May 19 2022])
```

Figure 16: SwitchExternal's alerts during ARP Poisoning

However, when instructed to ping Google, CameraTrap was unable to reach the switch to send packets. This is shown in Figure 17. The same result was obtained when CameraTrap attempted to ping PC1 (10.1.1.101).



```
CameraTrap> ping 8.8.8.8

host (10.1.2.1) not reachable
```

Figure 17: CameraTrap attempts to ping Google

It is uncertain why this occurred. As the switch provided alerts (in Figure 16) it is assumed that the mitigations were successfully implemented. If these same mitigations rendered the switch unreachable, there may be a flaw in the virtual hardware or an unknown software failure.

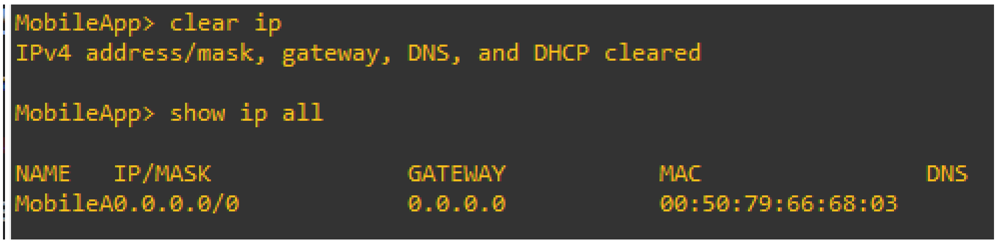
3.5.2. DHCP Spoofing

When new hosts are connected to an existing network, a DHCP ensures that the hosts receive the correct network configuration, including the appropriate DNS server and IP address, for example. Bhushan, Sahoo and Rai (2017) state that DHCP has a high standard of security, but that DHCP messages are not guaranteed to originate from trusted servers or legitimate hosts.

During a DHCP spoofing attack, an attacker typically specifies a false DNS server, a fake default gateway and a pool of incorrect IP addresses, to establish a rogue DHCP server. This server is deployed on a network and when a new host connects to the network, the host can unintentionally acquire its network configuration from the malicious server, enabling the attacker to capture the host (Bhushan, Sahoo & Rai, 2017).

Attack:

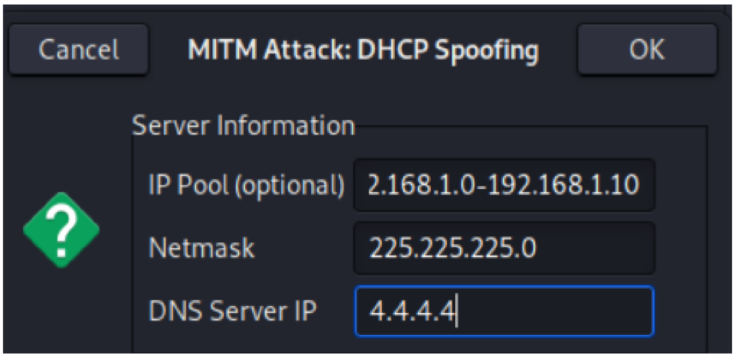
For this attack, a DHCP spoof was attempted on MobileApp (10.1.2.103) while using Ettercap. To proceed, the IP address of MobileApp was cleared and verified to be empty, as is shown in Figure 18.



```
MobileApp> clear ip
IPv4 address/mask, gateway, DNS, and DHCP cleared

MobileApp> show ip all

NAME   IP/MASK              GATEWAY           MAC                DNS
MobileA0.0.0.0/0            0.0.0.0           00:50:79:66:68:03
```

Figure 18: MobileApp's cleared IP address

In Figure 19, one can see the attack will be launched with a pre-specified IP pool of 192.168.1.0 - 192.168.1.10, a default subnet mask of 255.255.255.0 and a DNS server of 4.4.4.4. If the attack succeeds, one should see these values reflected in the area in Figure 18, after the “show ip all” command is run.



Server Information

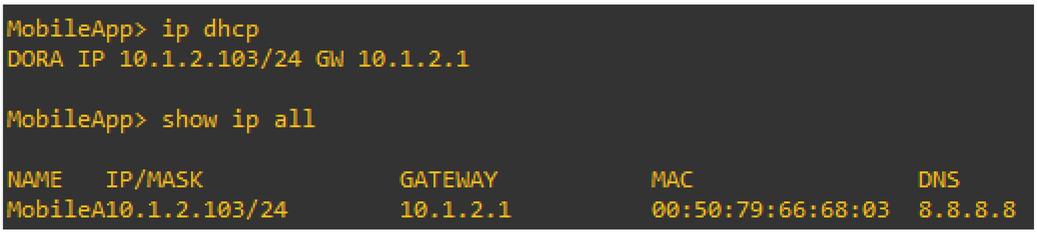
IP Pool (optional) 2.168.1.0-192.168.1.10

Netmask 225.225.225.0

DNS Server IP 4.4.4.4

Figure 19: DCHP spoofing attack setup

In Figure 20 one can see the result of the attack. After the IP address was re-initialised with “ip dhcp” on MobileApp, it is seen that the IP address assumed by MobileApp (10.1.2.103) is the same one that it had prior to being cleared. This indicates that MobileApp re-acquired an IP from the C3745 router’s DHCP server. This DCHP spoofing attack therefore failed. The attack was repeated and failed again.



```
MobileApp> ip dhcp
DORA IP 10.1.2.103/24 GW 10.1.2.1

MobileApp> show ip all

NAME   IP/MASK              GATEWAY           MAC                DNS
MobileA10.1.2.103/24        10.1.2.1          00:50:79:66:68:03  8.8.8.8
```

Figure 20: MobileApp's IP acquisition during DHCP spoofing

Mitigation:

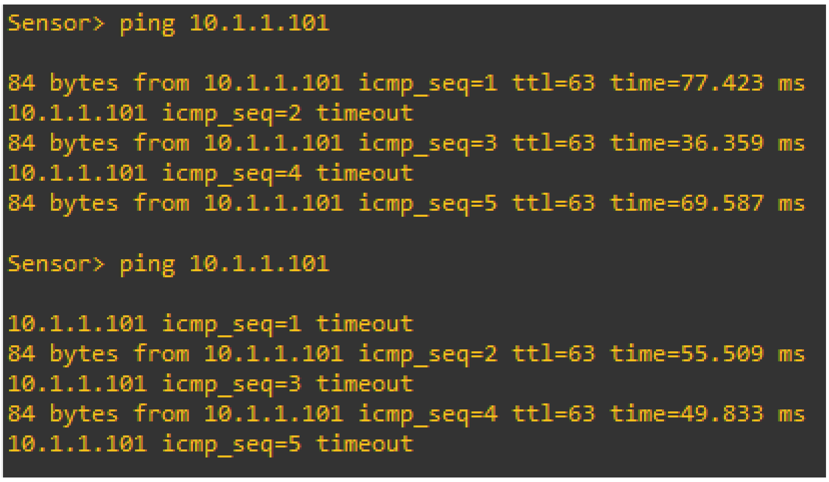
DCHP snooping can be applied to the switch as a countermeasure to the existence of a rogue DHCP server (Bhushan, Sahoo & Rai, 2017 and Bergs, *et al*., 2020). As the DHCP spoofing attack failed here, one will not be able to confirm that an applied mitigation is working. Therefore, DHCP snooping was not enabled in this instance. However, as it is enabled for other attacks, one should then still be afforded protection against DHCP spoofing.

3.5.3. DHCP Starvation

During a DHCP starvation attack, an attacker constantly directs counterfeit DHCP client requests to the network’s DHCP server, to deplete the server’s pool of available IP addresses. When the available IP pool eventually dries up, authentic clients are barred from obtaining IP addresses (Aldaoud, *et al*., 2021). This is a form of DoS attack.

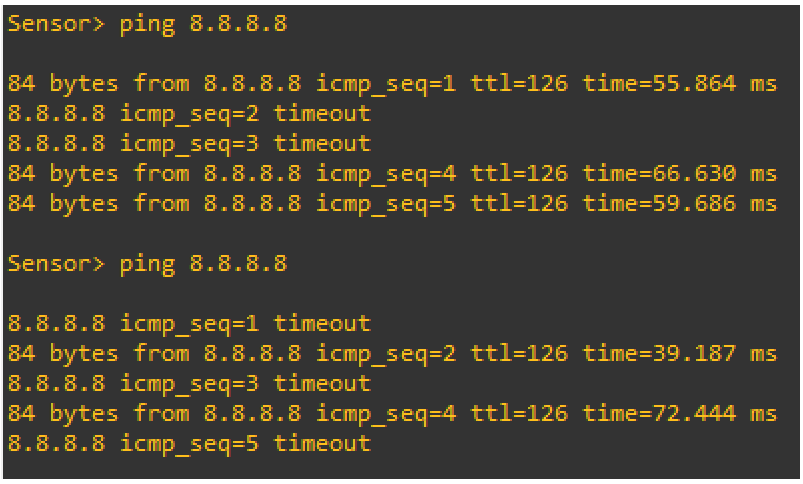
Attack:

The Yersinia tool on the Kali machine was used for this attack. In Yersinia, the DHCP starvation attack is launched by selecting to send “DHCP DISCOVER” packets to the switch. This should in theory slow the switch down and disrupt the intended target, Sensor’s (10.1.2.105), ability to communicate and send packets via the Internet and the internal network. By analysing the packet data when Sensor is commanded to ping 8.8.8.8 and PC1 (10.1.1.101), one can see that the attack is succeeding, as Sensor can only intermittently get through to each destination. The effect of the attack on Sensor can be seen in Figures 21 and 22.



```
Sensor> ping 10.1.1.101

84 bytes from 10.1.1.101 icmp_seq=1 ttl=63 time=77.423 ms
10.1.1.101 icmp_seq=2 timeout
84 bytes from 10.1.1.101 icmp_seq=3 ttl=63 time=36.359 ms
10.1.1.101 icmp_seq=4 timeout
84 bytes from 10.1.1.101 icmp_seq=5 ttl=63 time=69.587 ms

Sensor> ping 10.1.1.101

10.1.1.101 icmp_seq=1 timeout
84 bytes from 10.1.1.101 icmp_seq=2 ttl=63 time=55.509 ms
10.1.1.101 icmp_seq=3 timeout
84 bytes from 10.1.1.101 icmp_seq=4 ttl=63 time=49.833 ms
10.1.1.101 icmp_seq=5 timeout
```

Figure 22: Sensor attempting to ping PC1 during DHCP starvation



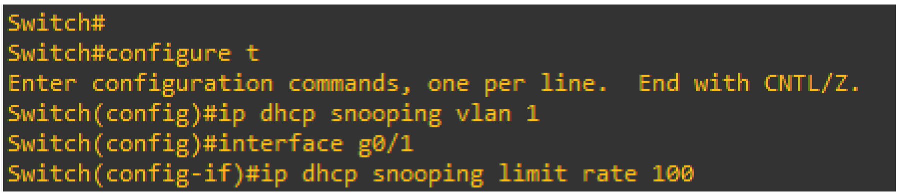
```
Sensor> ping 8.8.8.8

84 bytes from 8.8.8.8 icmp_seq=1 ttl=126 time=55.864 ms
8.8.8.8 icmp_seq=2 timeout
8.8.8.8 icmp_seq=3 timeout
84 bytes from 8.8.8.8 icmp_seq=4 ttl=126 time=66.630 ms
84 bytes from 8.8.8.8 icmp_seq=5 ttl=126 time=59.686 ms

Sensor> ping 8.8.8.8

8.8.8.8 icmp_seq=1 timeout
84 bytes from 8.8.8.8 icmp_seq=2 ttl=126 time=39.187 ms
8.8.8.8 icmp_seq=3 timeout
84 bytes from 8.8.8.8 icmp_seq=4 ttl=126 time=72.444 ms
8.8.8.8 icmp_seq=5 timeout
```

Figure 21: Sensor attempting to ping Google during DHCP starvation

Mitigation:

Bergs, *et al*., (2020) and Aldaoud, *et al*., (2021) state that enabling DHCP Snooping and rate limiting on a switch can counter the effects of a DHCP starvation attack. Cisco (n.d) and Abdulhafiz, *et al*., (2020) state that DHCP snooping can be activated on each VLAN (Virtual Local Area Network), as by default the feature is disabled. Cisco (n.d.) also mentions that rate-limiting is not applied out of the box and requires additional activation.

The following commands, seen in Figure 23, were entered on SwitchExternal (10.1.2.1).

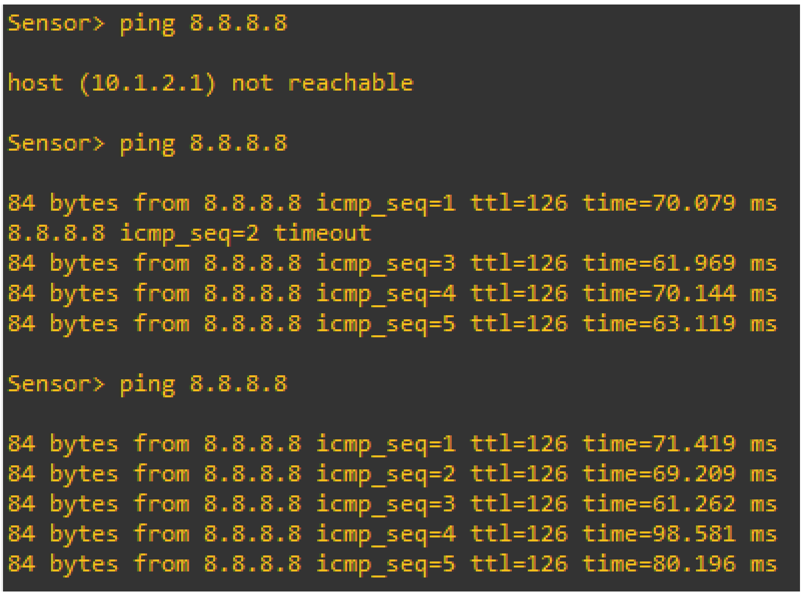


```
Switch#
Switch#configure t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#ip dhcp snooping vlan 1
Switch(config)#interface g0/1
Switch(config-if)#ip dhcp snooping limit rate 100
```

Figure 23: Setting DHCP snooping and rate limits on the switch

The limit rate was set to 100 on the interface of the Kali machine, as per the recommendation of Cisco (n.d.) for untrusted interfaces.

The attack was relaunched, and it is evident that effective mitigation occurred when viewing Figure 24, as Sensor could ping Google unimpeded after a few attempts.



```
Sensor> ping 8.8.8.8

host (10.1.2.1) not reachable

Sensor> ping 8.8.8.8

84 bytes from 8.8.8.8 icmp_seq=1 ttl=126 time=70.079 ms
8.8.8.8 icmp_seq=2 timeout
84 bytes from 8.8.8.8 icmp_seq=3 ttl=126 time=61.969 ms
84 bytes from 8.8.8.8 icmp_seq=4 ttl=126 time=70.144 ms
84 bytes from 8.8.8.8 icmp_seq=5 ttl=126 time=63.119 ms

Sensor> ping 8.8.8.8

84 bytes from 8.8.8.8 icmp_seq=1 ttl=126 time=71.419 ms
84 bytes from 8.8.8.8 icmp_seq=2 ttl=126 time=69.209 ms
84 bytes from 8.8.8.8 icmp_seq=3 ttl=126 time=61.262 ms
84 bytes from 8.8.8.8 icmp_seq=4 ttl=126 time=98.581 ms
84 bytes from 8.8.8.8 icmp_seq=5 ttl=126 time=80.196 ms
```

Figure 24: Sensor pinging Google successfully

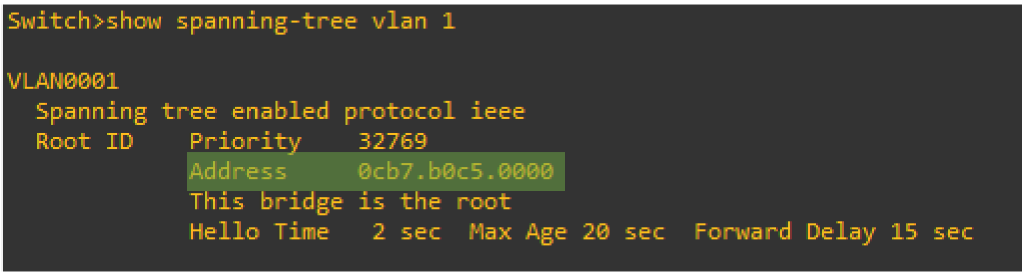
3.5.4. STP Attack

A Spanning Tree Protocol (STP) is implemented on a network's switches, to attain a logical topology that is loop free, based on a physical topology where loops are present. This is beneficial, as physical loops pose a risk to the network (e.g., table corruption) and redundant topologies offers easier management to network administrators. STP uses a Bridge Protocol Data Unit (BPDU), containing all the necessary fields to determine if a newly received BPDU supplants a stored BPDU. The BPDU with the highest priority wins the "election" to become the STP root (Trejo, Monroy & Monsalvo, 2006).

However, since STP has no authentication processes, a BDPU can be deployed on the network by an attacker, which specifies the highest priority. Received as a legitimate unit, the STP will now permit the attacker to win root elections, which results in the attacker becoming the new root of the network (Trejo, Monroy & Monsalvo, 2006).

Attack:

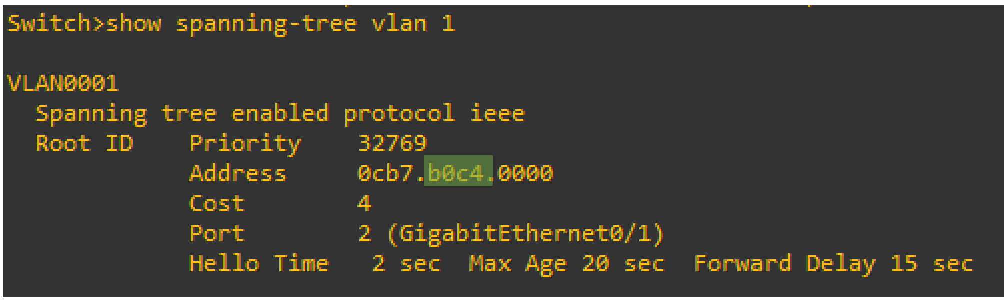
The “show spanning-tree vlan 1” command, entered on SwitchExternal’s console, displays the Root ID details of VLAN0001, and can be seen in Figure 25. The Root ID Address is the important part to take note of:



```
Switch>show spanning-tree vlan 1

VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
             Address     0cb7.b0c5.0000
             This bridge is the root
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
```

Figure 25: VLAN 1 spanning tree and root ID address

The goal is to replace the spanning bridge address with that of the Kali machine. Yersinia was the selected tool. By launching an STP attack and instructing Yersinia to “Claim Root Role”, the result should be an altered address. In Figure 26 it is evident that the attack succeeded, as the middle part of the address (highlighted in green) changed from “b0c5” in Figure 25 to “b0c4” in Figure 26, and the port now matches that of the Kali machine.

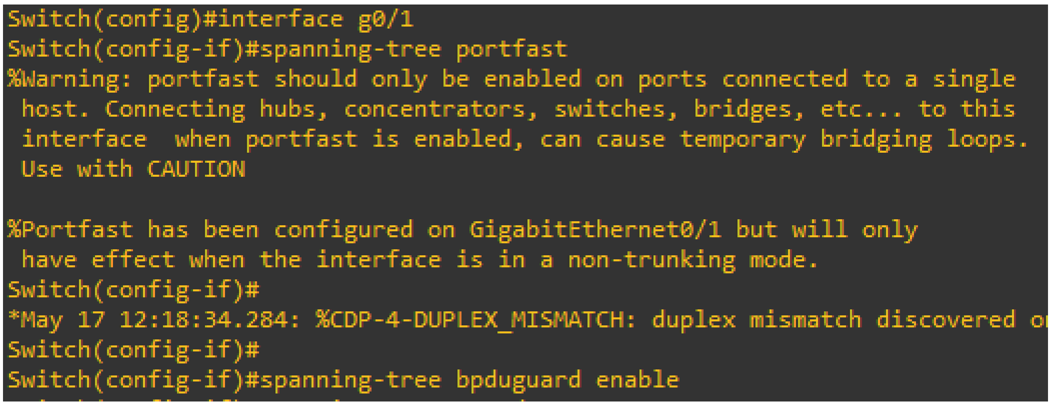


```
Switch>show spanning-tree vlan 1

VLAN0001
  Spanning tree enabled protocol ieee
  Root ID    Priority    32769
             Address     0cb7.b0c4.0000
             Cost        4
             Port        2 (GigabitEthernet0/1)
             Hello Time   2 sec  Max Age 20 sec  Forward Delay 15 sec
```

Figure 26: Captured spanning tree on VLAN 1

Mitigation:

Bergs, *et al.*, (2020) specify portfast, bpduguard and “guard root” as the main mitigators for the STP attack. Teofilo (n.d.) details how to implement portfast, bpduguard and “guard root”. In Figure 27, one can see that portfast, bpduguard, and “guard root” has been enabled on SwitchExternal. The switch does alert the user to the dangers of using portfast without the mitigation of bpduguard.



```
Switch(config)#interface g0/1
Switch(config-if)#spanning-tree portfast
%Warning: portfast should only be enabled on ports connected to a single
 host. Connecting hubs, concentrators, switches, bridges, etc... to this
 interface  when portfast is enabled, can cause temporary bridging loops.
 Use with CAUTION

%Portfast has been configured on GigabitEthernet0/1 but will only
 have effect when the interface is in a non-trunking mode.
Switch(config-if)#
*May 17 12:18:34.284: %CDP-4-DUPLEX_MISMATCH: duplex mismatch discovered o
Switch(config-if)#
Switch(config-if)#spanning-tree bpduguard enable
```

Figure 27: STP attack mitigations applied to switch

After the mitigations have been implemented, SwitchExternal’s STP address is firstly confirmed to be that of Figure 25. Using Yersinia to attack again with “Claim Root Role”, it was verified that the mitigations were proven to be effective, as Kali could not claim the root role and the root address remained unaltered.

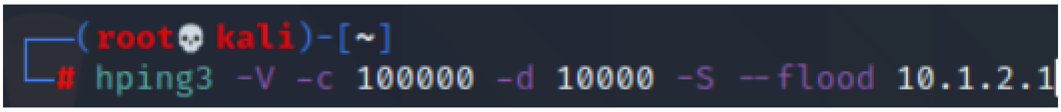
3.5.5. TCP/SYN Flood

The Transmission Control Protocol (TCP) is a central protocol in the set of Internet protocols, says Gavaskar, *et al.*, (2010). TCP is only interested in two end systems, such as a Web server and Web browser. TCP is responsible for the delivery of packets from a Program X on Computer A to Program Y on Computer B, and this can include file transfers and e-mails. TCP works based on a three-way handshake. This means that a third packet confirms that the originator can receive packets at its source request’s IP address, or that it is accessible for return packets. Gavaskar, *et al.*, (2010) elaborates that TCP can control the packet size, flow, and rate of data exchange.

SYN flooding attacks "flood" a network with the intention of exhausting resources, such as memory and bandwidth, of a target. Therefore, it is classified as a DoS attack. This attack is done by continuously sending seemingly legitimate SYN requests, that contain a falsified source IP address, to a target host. This causes the target system to suffer numerous half-open connections, which results in the system's inability to create new connections, leaving the target service "down" (Gavaskar, *et al*., 2010).

Attack:

A TCP/SYN Flood attack was also attempted on the network, with SwitchExternal (10.1.2.1) as the target. The Kali programmes used here was hping3 and Wireshark. In Figure 28, the code command is shown to initialise the flood attack from Kali's terminal. While the attack is taking place, Wireshark can be used as a packet analyser to inspect the network traffic.

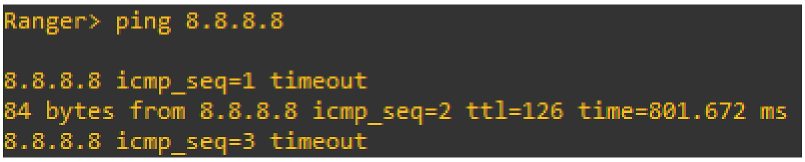


```
┌──(root💀kali)-[~]
└─# hping3 -V -c 100000 -d 10000 -S --flood 10.1.2.1
```

Figure 28: hping3 TCP/SYN flood attack

However, Wireshark presented a problem. Wireshark would repeatedly fail to fully initialise in the Kali VM and its use was abandoned, and this problem is further described in this chapter's limitations.

Even though the packets could not be inspected, the effect of hping3's attack can be tested by analysing a component. While running the attack, hping3 does not provide any error messages and instead indicates that it is in "flood mode". According to Bergs, *et al*., (2020), the expected effect on a device would be an increase in ping response times. When Ranger (10.1.2.102) was instructed to ping Google, seen in Figure 29, the responses timed out or had very high latencies. However, no effect was observed when pinging an internal network device, nor was the result obtained by Bergs, et al. (2020) observed, where the authors pinged the switch and received high latency.



```
Ranger> ping 8.8.8.8

8.8.8.8 icmp_seq=1 timeout
84 bytes from 8.8.8.8 icmp_seq=2 ttl=126 time=801.672 ms
8.8.8.8 icmp_seq=3 timeout
```

Figure 29: Ranger attempting to ping Google during TCP/SYN flood

Mitigation:

Bergs, *et al.*, (2022) mitigated a TCP/SYN Flood attack by implementing a firewall on their proposed network. While GNS3 is an open-source platform, certain network appliances are situated behind a proprietary paywall. Therefore, it will only be acknowledged here that in theory a firewall should mitigate the attack, but that it will not be implemented in practice to verify this on the present network.

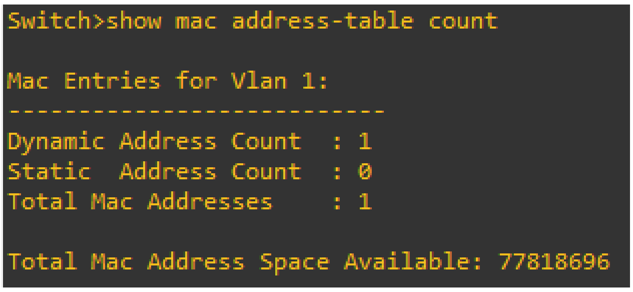
3.5.6. CAM Overflow

Mahmood, *et al.*, (2020), describe that a Content Addressable Memory (CAM) table is a system constructed memory table. As Cisco switches record the end-users MAC (Media Access Control) address, associated ports and VLAN ID in CAM tables, these switches are vulnerable to CAM overflow attacks.

Typically, a Cisco CAM table can concurrently store thousands of MAC addresses. If an attacker constantly sends fake MAC addresses to the CAM table, the table "overflows", where its reserved space is filled up. This in turn converts the target switch into a hub which gives the attacker the ability to access every client in the VLAN. This enables attackers to extract information and perform further attacks more efficiently (Mahmood, *et al.*, 2020).

Attack:

First, the MAC address table needs to be inspected on the target switch, SwitchExternal. Proceeding to enter "show mac address-table count" provides one with the information depicted in Figure 30. It is important to note here that the switch still has MAC address space available.



```
Switch>show mac address-table count

Mac Entries for Vlan 1:
---------------------------
Dynamic Address Count  : 1
Static  Address Count  : 0
Total Mac Addresses    : 1

Total Mac Address Space Available: 77818696
```

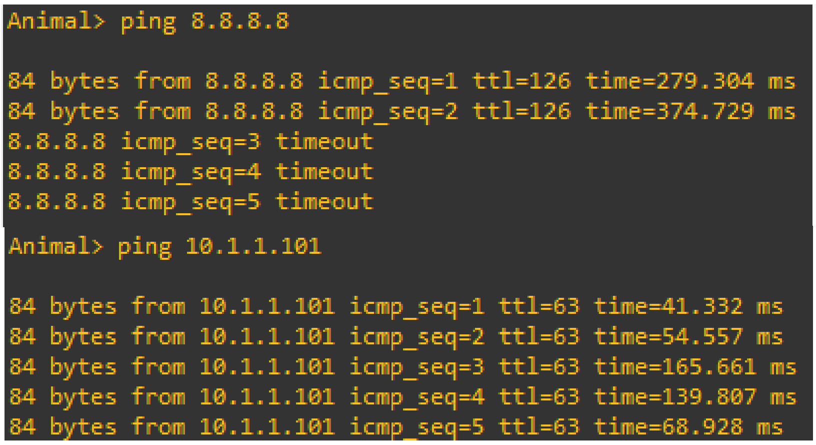
Figure 30: MAC address table on the switch

Kali's built-in "macof" tool was used to launch the CAM Overflow attack. A simple line of code, "macof -i eth0" creates a serious attack. When the code is run, multitudes of MAC addresses are continuously generated and appear in the terminal. To kill the process, one must enter Control + C, otherwise the attack does not stop.

Theoretically, the CAM Overflow attack should now have taken up all the switch's previously available space. To test this, the table is inspected again from the attacked switch's console, where one expects to find a zero value for "Total Mac Address Space Available".

This was not the case. The switch completely went down and ceased to function. Nothing could be entered into the console. Only when the attack was stopped did the switch, with some latency, start to perform the commands entered while the attack was in progress.

While the attack was in progress, a field device, Animal, was analysed, with the results shown in Figure X. When attempting to ping Google, the packet transfer either timed out or had a higher latency than usual. When pinging a JOC device, such as PC1 (10.1.1.101), Animal was able to transfer packets without timeouts, but some transfers also presented with a higher latency.

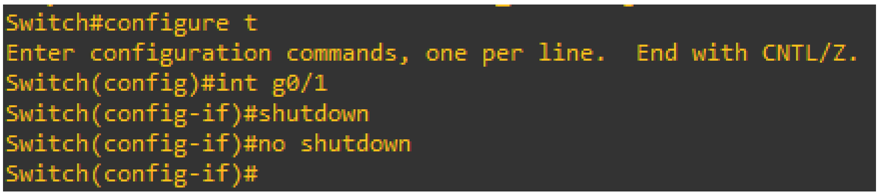


```
Animal> ping 8.8.8.8

84 bytes from 8.8.8.8 icmp_seq=1 ttl=126 time=279.304 ms
84 bytes from 8.8.8.8 icmp_seq=2 ttl=126 time=374.729 ms
8.8.8.8 icmp_seq=3 timeout
8.8.8.8 icmp_seq=4 timeout
8.8.8.8 icmp_seq=5 timeout
Animal> ping 10.1.1.101

84 bytes from 10.1.1.101 icmp_seq=1 ttl=63 time=41.332 ms
84 bytes from 10.1.1.101 icmp_seq=2 ttl=63 time=54.557 ms
84 bytes from 10.1.1.101 icmp_seq=3 ttl=63 time=165.661 ms
84 bytes from 10.1.1.101 icmp_seq=4 ttl=63 time=139.807 ms
84 bytes from 10.1.1.101 icmp_seq=5 ttl=63 time=68.928 ms
```

Figure 31: Animal's pinging success during CAM overflow attack

Mitigation:

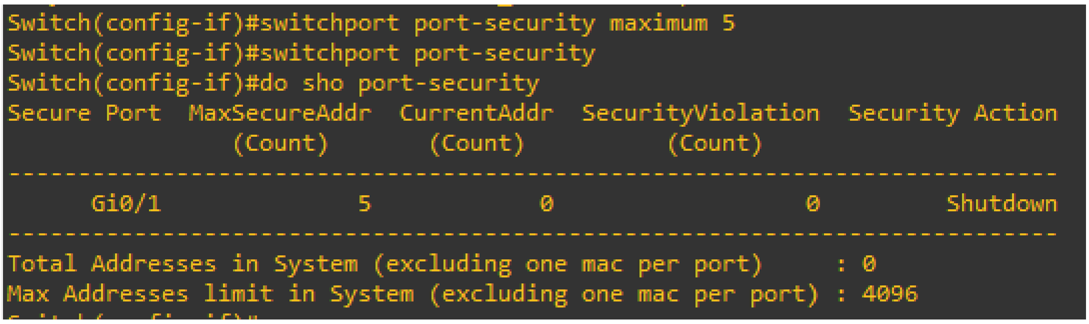
According to Heintzkill (2021), the best way to prevent a CAM Overflow Attack is to enable port security on the vulnerable switch. The author details how this may be achieved. In Figure 32, it shows how the switch was firstly "bounced". This means to shut it off and turn it back on to clear all MAC addresses saved on the port leading to the Kali machine.



```
Switch#configure t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#int g0/1
Switch(config-if)#shutdown
Switch(config-if)#no shutdown
Switch(config-if)#
```

Figure 32: "Bouncing" the switch

After commanding the switch to run "switchport mode access", port security was enabled as seen in Figure 33. "maximum 5" indicates that only five MAC addresses can be received on that port, as per Heintzkill's (2021) recommendation. Next, the port-security was activated, and it was shown to be active on the Kali port in the displayed table. It shows that no more than five MAC addresses will be accepted and the action it will take is to shut down the port if more than that are sent to the port.



```
Switch(config-if)#switchport port-security maximum 5
Switch(config-if)#switchport port-security
Switch(config-if)#do sho port-security
Secure Port  MaxSecureAddr  CurrentAddr  SecurityViolation  Security Action
                (Count)       (Count)          (Count)
---------------------------------------------------------------------------
      Gi0/1              5             0                  0         Shutdown
---------------------------------------------------------------------------
Total Addresses in System (excluding one mac per port)     : 0
Max Addresses limit in System (excluding one mac per port) : 4096
```

Figure 33: Applying mitigations against CAM overflow attacks on the switch

After the CAM Overflow Attack was launched again, SwitchExternal was now able to withstand the attack and continued to function. Animal (10.1.2.100) was then again instructed to ping Google and PC1. Now Animal could send and receive packets with normal latency and no timeouts on both connections. Therefore, it is evident that the implemented port security measures were enough to curb the attack.

3.5.7. VLAN Hopping

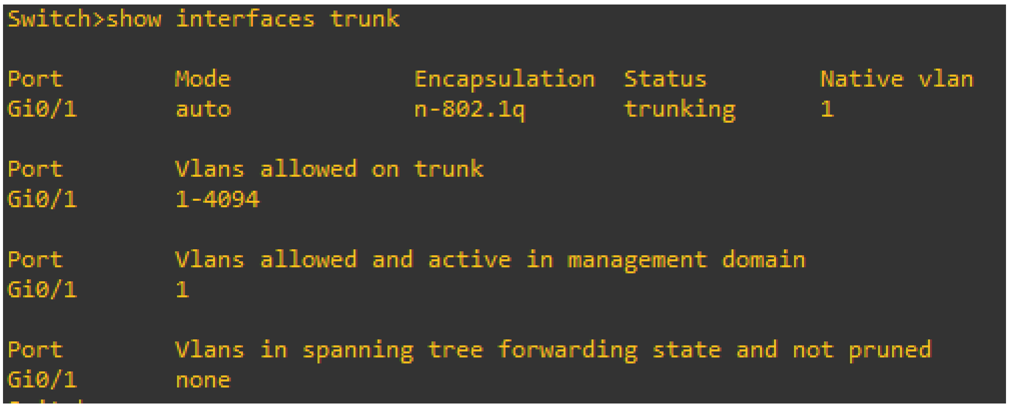
In a VLAN, a root bridge eases traffic flow between switches and access links connect users to their specific VLAN. The presence of open ports on a VLAN allows new connections, but anyone can connect to these ports (Mahmood, *et al*., 2020).

Cisco has developed a dynamic trunking protocol (DTP) to facilitate the detection of trunk links between switches. The DTP can handle new trunk links and detect the utilised encapsulation (Mahmood, *et al*., 2020).

During a VLAN hopping attack, access links are converted into trunk links by attackers sending forged DTP messages through the VLAN. The attacker can now access any network traffic that was previously filtered from the access links and can therefore view all communications on that trunk link (Mahmood, *et al*., 2020).

Attack:

A VLAN hopping attack was performed. First SwitchExternal is inspected and commanded to run "show interfaces trunk". This command returns no output, as there should be no trunk present at this stage. In Yersinia, a DTP attack was launched, with “enable trunking” specified as the main attack method. Upon reinspection of the switch, running "show interfaces trunk" now returns output, showcasing the trunks, as seen in Figure 34.

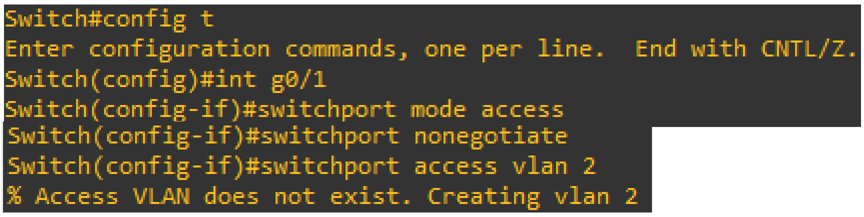


```
Switch>show interfaces trunk

Port        Mode             Encapsulation  Status        Native vlan
Gi0/1       auto             n-802.1q       trunking      1

Port        Vlans allowed on trunk
Gi0/1       1-4094

Port        Vlans allowed and active in management domain
Gi0/1       1

Port        Vlans in spanning tree forwarding state and not pruned
Gi0/1       none
```

Figure 34: Trunk interfaces on SwitchExternal during VLAN hopping

Mitigation:

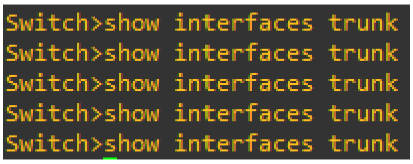
Popeskic (2013) provides guidelines for countering VLAN Hopping. The steps include disabling trunking, and DTP prevention. These steps are shown in Figure 35.



```
Switch#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#int g0/1
Switch(config-if)#switchport mode access
Switch(config-if)#switchport nonegotiate
Switch(config-if)#switchport access vlan 2
% Access VLAN does not exist. Creating vlan 2
```

Figure 35: Mitigating VLAN hopping on the switch

When Yersinia is again instructed to launch the VLAN hopping attack, it does not work this time. Figure 36 shows how the switch was prompted several times to show trunks, but had no output to show, indicating that the mitigation was effective.



```
Switch>show interfaces trunk
Switch>show interfaces trunk
Switch>show interfaces trunk
Switch>show interfaces trunk
Switch>show interfaces trunk
```

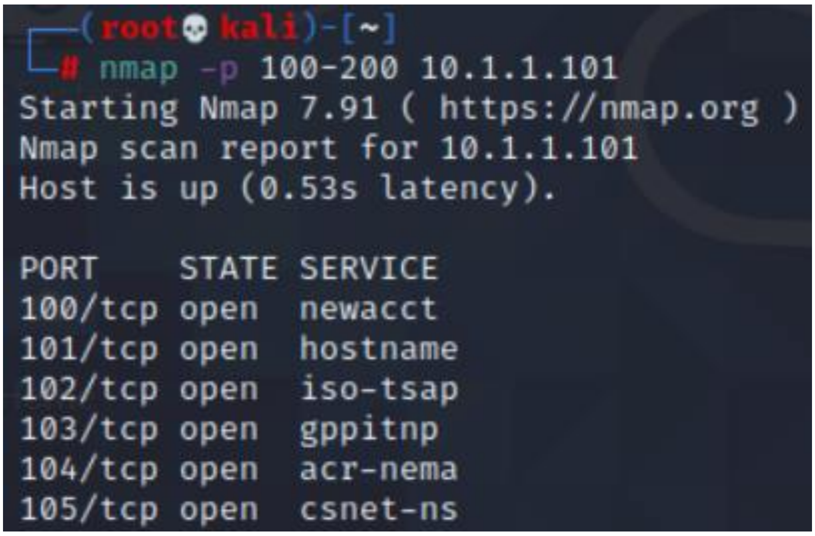
Figure 36: Switch presenting with no trunks

3.5.8. Port Scanning

Shah, *et al*., (2019) states that any Internet connected device has ports, that each run a particular service. For example, port 80 is typically associated with “http”. Ports can be open or closed, depending on the needs of the host device. A message can be relayed to various ports to determine which are open, and this type of scouting is known as port scanning. Gavaskar, *et al*., (2010) state that ports are scanned to elicit a response, which will identify the responding port as a potential service to exploit. According to Shah, *et al*., (2019), port scanning is mostly performed without malicious intent, but naturally there exists attackers that want to exploit open ports.

Attack:

Figure 37 shows that Nmap was instructed to scan ports 100-200 (a random sample range) on the target machine, PC1 (10.1.1.101). Nmap determined that each of those ports were open, and provided the estimated service associated with a specific port. Figure 37 displays just a snapshot of the first few open ports.



```
┌──(root💀 kali)-[~]
└─# nmap -p 100-200 10.1.1.101
Starting Nmap 7.91 ( https://nmap.org )
Nmap scan report for 10.1.1.101
Host is up (0.53s latency).

PORT    STATE SERVICE
100/tcp open  newacct
101/tcp open  hostname
102/tcp open  iso-tsap
103/tcp open  gppitnp
104/tcp open  acr-nema
105/tcp open  csnet-ns
```

Figure 37: Ports determined by Nmap to be open on PC1

Mitigation:

Barnett (2019) states that a firewall, with customisable port configuration, can help to limit the incoming and outgoing traffic on a device or internal network. Due to software limitations, the effect of a firewall mitigation was not applied to the port scanning attack at this time. The attack was solely performed to showcase the default existence of open ports which can translate into vulnerabilities.

3.6. Threat Modelling

Threat modelling can now be performed, as per Gonzalez' (2022) instructions. Threat data has been gathered, the assets in an antipoaching system have been identified and mitigations were applied. The next step is to assess risk and outline threats, which will be attempted with the threat models in this and the following chapter.

3.6.1. Attack Trees

Schneier (1999) set out to find a way in which threats can be modelled against computer systems. The logic was that computer security is frequently bypassed, therefore if threats can be pre-empted, effective mitigations may be developed. Schneier (1999) decided that attack trees were the solution, a formal approach that sees the attacking goal as the root node and the leaf nodes are the different methods an attacker can use to achieve their goal. An example of a visual attack tree can be seen in Figure 38, but attack trees in text form are also a possibility.

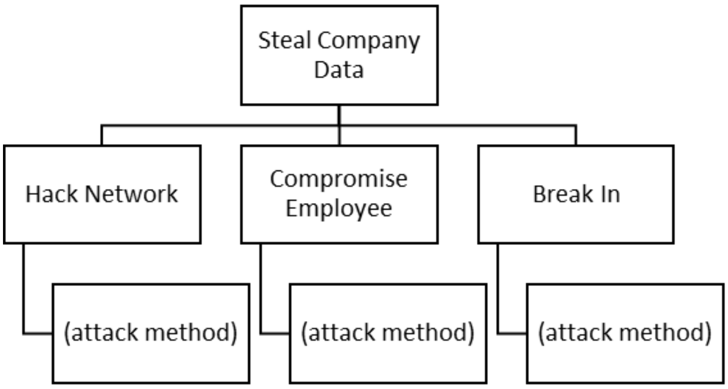


Figure 38: Example Attack Tree

Schneier (1999) expands on the idea of the attack tree by incorporating AND / OR nodes, and the assignment of Possible (P) or Impossible (I) to each node. It is then easier to see if an attack can be carried out, as all nodes leading to the goal must be possible, and the attack is deemed impossible otherwise. The attack tree can be altered to specify monetary amounts or equipment associated with each node or attack method. The opposite of attack trees, defence trees, can also be modelled.

The benefit of attack trees is that one can see not just the potential threats, but also the possible sources of these threats.

3.6.2. STRIDE

Anwar, Nazir, and Ansari (2020) describe STRIDE as a practical and accessible threat modelling process. STRIDE is an acronym that encapsulates six major threat categories, and Table 5 provides the descriptions of them (Abomhara, Gerdes & Køien, 2015).

Table 5: STRIDE description

Threat Class	Description
Spoofing	Assuming a false identity to try and gain access to a system
Tampering	The alteration or destruction of data without proper authority

Repudiation	The denial by users of certain events or transactions in the system
Information Disclosure	Occurs when data is undesirably exposed
Denial of Service	An action that leads to the unavailability of a service or system
Elevation of Privilege	Occurs when a user gains access to a higher level of privilege than is authorised

Shostack (2014) notes that the aim of STRIDE is to find attacks, not to categorise them. This means that if you are unsure about the exact category of a newly discovered threat, it does not matter as long as the threat is recorded. The classification of threats may only be useful in relation to formulating mitigations, says Shostack (2014).

3.6.3. DREAD

The DREAD threat model allows one to identify and assess intruder threats (Pevnev, *et al.*, 2021). This model works by scoring threats (subjectively) and using these scores to evaluate risk based on a predefined scale (Zhang, *et al.*, 2021). Depending on the threats assessed, one can develop a safety profile that spans various categories of the system (Pevnev, *et al.*, 2021).

An acronym, “DREAD” is short for five risk attributes (Zhang, *et al.*, 2021):

- Damage Potential: How much damage can a threat cause?
- Reproducibility: Can the threat be easily duplicated by others?
- Exploitability: How easy is it to exploit the threat?
- Affected Users: Internally or externally, how many users will be impacted by the threat?
- Discoverability: Can one easily detect the threat?

With the original DREAD model, any given threat is scored between 1-3 per risk attribute in the model’s acronym, and the scores are added up. The total score value assigned to a threat is then compared to DREAD’s rating’s scale to determine the risk posed by the threat (Logix Consulting, 2019). Table 6 shows the correspondence between a score rating for each component of DREAD and the total score with low, medium, or high risk. Omotosho, Haruna and Olaniyi (2019) provide a classic DREAD table (Appendix B, Table 19) with which one can benchmark threats against, based on the 1, 2, 3 rating system.

Table 6: DREAD risk scores and levels

Assessed Risk	Low	Medium	High
Individual Risk Attribute	1	2	3
Total Threat Score	5-7	8-11	12-15

To better understand the traditional DREAD model, an example case study will be assessed.

Example Case Study (Threat): An attacker cracks a JOC worker's password (assuming no MFA) and stealthily retrieves a rhino's location. The DREAD assessment, based on Naagas and Palaoag (2018) (Appendix B, Table 20) and Omotosho, Haruna and Olaniyi (2019) (Appendix B, Table 19), for this scenario is shown in Table 7:

Table 7: Example case study's DREAD scoring

DREAD Item	Score	Explanation
Damage Potential	3	Secure data stolen; a rhino may be poached
Reproducibility	2	If cracked once, can crack again, but not easily
Exploitability	2	Success chance is estimated as medium
Affected Users	1	Can only access specific user's information
Discoverability	2	Depending on attacker's actions, may be difficult to discover
Total	10	Risk is medium – high (with reference to Table 6)

The above case study shows that even though the damage caused by a threat may be at a maximum, the other factors associated with the threat reduces the overall threat risk to a slightly lower degree. The DREAD scoring system therefore provides an overall indication of a threat's risk profile, but individual attribute scores can assist in prioritising various threats.

Per Logix Consulting (2019), DREAD enjoys widespread use as a threat model, and Pevnev, *et al*., (2021) names the DREAD model's ability to gain a high level of abstraction as an advantage of the model. However, Naagas and Palaoag (2018) and Zhang, *et al*., (2021) acknowledge that DREAD's subjective nature requires validation and present a novel method with which this can be achieved.

Zhang, *et al*., (2021) determine that all five of DREAD's risk attributes must be assessed to determine the risk of a threat. Naagas and Palaoag (2018) decided to assign the values of 0, 5 and 10 to a threat in lieu of the terms low, medium, and high respectively. A threat can be represented as (D, R, E, A, D) where $D, R, E, A, D \in \{0, 5, 10\}$, and where rs_t denotes the risk score as a quantified average of the five risk attributes' scores (Zhang, *et al.*, 2021). Naagas and Palaoag (2018) proposed Equation 3 for the original DREAD model:

$$rs_t = (D + R + E + A + D)/5$$

Equation 3: DREAD risk calculation by Naagas and Palaoag (2018)

Zhang, *et al.*, (2021) proposed a similar equation, but failed to include an explicit interpretation of the resultant risk score in their study. Naagas and Palaoag (2018) also did not include an explanation, but the authors applied their equation to data and one can infer from there what category the risk scores belong to. This equation can therefore be used to quantify risk, and the authors' scoring of risks is seen in Appendix B, Table 20.

To improve upon DREAD, Zhang, *et al*. (2021) decided to redefine some of the DREAD risk attributes to better fit their specific study topic of digital data marketplaces (DDMs). Zhang, *et al*., (2021) provide a table (Appendix B, Table 21) with qualitative descriptions for what low, medium, and high risk corresponds to per redefined risk attribute. In doing so, the authors provide an objective benchmark against which the subjective risk assessments of their improved DREAD can be made. As this table was specifically formulated with DDMs in mind, it is not entirely suited to the research landscape of this study, but this table provides the potential for adaptation.

Per Zhang, *et al*., (2021), their new DREAD model is more stable regarding subjective choices, but they caution that this is for a specific use case, and their use case was not anti-poaching systems. Therefore, more research may be needed to standardise the proposed improved DREAD model for more generic use cases.

In Table 8, one can view and adapted version of Omotosho, Haruna and Olaniyi's (2019), and Zhang, *et al*.'s (2021) benchmark tables. This new table is proposed as an interpretation of the classic DREAD model for use in anti-poaching system threat modelling. Further, it is proposed to modify D's definition to include "Poaching Potential" – the risk of a breach translating into a poaching incident. This addition was made to emphasise the potentially lethal (to both animals and rangers) consequences of an anti-poaching system breach.

Table 8: Adapted DREAD risk scoring table

Risk	D	R	E	A	D
Low (0)	Attacker can access trivial data Low poaching probability	Hard to reproduce	Advanced skills, difficult to exploit, takes time	One or two users	Likely detection chance, even without monitoring
Medium (5)	Attacker can access sensitive data Medium poaching probability	Somewhat reproducible	Intermediate skills, medium difficulty exploit, can take time	The JOC personnel	Moderate detection chance aided by monitoring
High (10)	Attacker gains full authorisation High poaching probability	Easily Reproducible	Amateur skills, easy exploit, done in short time	All anti-poaching personnel	Unlikely detection chance even if monitored

Considering this new DREAD, if we take the example case study from above:

Example Case Study (Threat): An attacker cracks a JOC worker's password (assuming no MFA) and stealthily retrieves a rhino's location. Table 9 shows this author's DREAD assessment, when cross-referenced with Table 8.

Table 9: Improved DREAD example case study scoring

Risk Level	**D**	**R**	**E**	**A**	**D**
Low (0)				X	
Medium (5)		X	X		
High (10)	X				X
D: Can access network and animal data, may cause high damage					
R: Reproducible, but may not be successful					
E: Some skill needed to perform attack					
A: In this case, affects one user, but has potential to recreate attack on other users.					
D: If attacker remains under radar, can be difficult to detect.					
Total: (10+5+5+0+10)/5 = 30/5 = 6 (Medium)					

Taking the above risk scoring into consideration, the average risk level for this threat is estimated to be medium per the new DREAD table. This differs from the slightly higher risk scored using the old DREAD model. The reason for this can be that the different definitions and benchmarks create a subtle difference in the score. It is again acknowledged that this is still subjective and should ideally be performed by a person with adequate knowledge and context of each threat.

This study will proceed to utilise the newer DREAD definitions and temper the benchmarks with information sourced from other scholars, such as Naagas and Palaoag (2018), and Omotosho, Haruna and Olaniyi (2019).

There is another point that can be made regarding the new DREAD estimation. The nature of threat modelling is to provide a factual summary of a threat's profile, but in doing so it can leave one with a lower risk level than one subjectively expects or would like, in this case due to the sentimentality that one can associate with the outcome of a threat. In the case study, it is a considerable loss to have an animal poached, but even with a high damage and poaching potential, the overall risk was calculated at a medium risk level.

3.7. Password Security

The CERN Computer Security Team put together a list of the characteristics of a good and bad password's makeup, as well as provide recommendations and examples for password creation (CERN, 2018). This can be viewed in Table 10 and Table 11.

Table 10: Good and bad password characteristics by CERN (2018)

Good Password	Bad Password
Known only to one person	Contains the username in any form
Remains secret, is not written down	Contains personal information, such as names or pets
Memorable	Contains numbers associated with you, such as license plates or phone numbers
Contains 8 or more characters	Contains generic sequences such as 'abcdef' or 'qwerty'
Contains a mix of 3 of uppercase, lowercase, digit, and symbol characters	Is reused for multiple accounts
Does not appear in a dictionary in any language	Contains dictionary listed words or abbreviations and acronyms
A program should not be able to guess it in a practical amount of time	Remains unchanged over a long period of time

Table 11: Password recommendations with examples by CERN (2018)

Password Recommendations	Example
Choose a line from a song or a phrase and use only the first letters of each word.	"I bless the rains down in Africa!" becomes "IbtrdiA!"
Use a long (memorable) phrase	"IsThisTheRealLife?IsThisJustFantasy?"
Use a mathematical formula	a^2+b^2=c^2
Create nonsense words, alternating between vowels and consonants. (these words are easy to pronounce and are more memorable). Mix in digits and symbols	"PakoLuki2" "Feri-Hano" "AlopBigu!"
Split one long word, or use shorter words and join them with punction character(s)	"Ctrl!Alt!Del!3" "Cyber/\Security"

3.7.1. Multi-Factor Authentication (MFA)

According to Ion, Reeder and Consolvo’s (2015) survey, there are concerns surrounding the availability or accessibility of MFA. These concerns regard the understanding of users of MFA, the usability of MFA and the feasibility of MFA in many instances.

Ometov, *et al.*’s (2019) study of MFA methods, specifically for IoT devices, yield the list of methods seen in Appendix D, Table 22. Further investigation by Ometov, *et al*., (2019) into the security, usability, and robustness of several of these methods (Appendix D, Table 23) can aid in determining which methods are applicable and more suitable for an anti-poaching system.

If it is assumed that a password will be a given authentication mechanism for personnel in an anti-poaching system, one or more additional mechanisms is required to satisfy MFA. Table 12 lists the authentication mechanisms (selected from Ometov, *et al*., 2019 & Ali, Dida, & Sam, 2020) considered to be the most feasible for an anti-poaching system. Biometric authentication is a popular alternative to PINs or passwords, and Almehmadi and Alsolami (2019) have found that nearly two thirds of their research respondents prefer using biometrics over passwords.

Table 12: Description of authentication mechanisms (Ometov, *et al*., 2019 & Ali, Dida, & Sam, 2020)

Authentication Mechanism	**Description**
Password	Familiar to most people. If not created “strong”, can be cracked
PIN code	Usually a short code, may be easier to crack than a password
One-Time-PIN (OTP)	An OTP is a unique, once off code. It is only received by a user upon request or as an added login requirement. Expires after use
Token	Physical item. Can be lost or stolen, not feasible for all purposes
Fingerprint Recognition	Biometric. Can be implemented for access control, system verification and on mobile phones
Facial Recognition	Biometric. Can be implemented for access control, system verification and on mobile phones
Geographical location	Can help prevent external attacks. The JOC typically remains stationary and could make using this authentication effective. Rangers on the move may have issues with this type of authentication

Gope and Sikdar (2019) developed an MFA scheme specific to IoT devices, making use of physically unclonable functions. While not within the scope here to detail this scheme in its entirety, Gope and Sikdar's (2019) scheme generally proved impervious to attacks and is computationally optimised. As an anti-poaching system usually contains one or multiple IoT devices, such schemes can prove beneficial and should be a consideration.

In relation to the risks, or threats, faced by an anti-poaching system, Ali, Dida, and Sam (2020) performed an investigation into the threat models on authentication and how this may be countered. Table 13 below shows the extracted information deemed relevant here, based on the common threats identified so far.

Table 13: Threats and authentication countermeasures (Ali, Dida & Sam (2020)

Threat	**Authentication Countermeasure**
Spoofing	Biometric
Phishing	MFA
Trojan	Biometric
MitM attack	Biometric (fingerprint) and MFA
DoS attack	Biometric (fingerprint)

Based on the above information, MFA, including a biometric factor, appears to be an essential requirement when faced by common security threats.

3.7.2. Password Managers

Almehmadi and Alsolami (2019)'s survey indicates that 9 in every 10 participants have never used a password manager, and that they may even be unaware of such software and their importance in maintaining password security.

There are many benefits of password managers. They can generate random text strings for use as passwords, thus creating a strong and unique password for each individual account. Password managers store passwords for specific websites, therefore they can shield one from entering passwords into malicious sites, as the password manager will not recognize a phishing site as the legitimate site it has registered a password on (CERN, 2018).

In Habib, *et al*.'s (2018) survey on password behaviour, the authors recommend that enterprise password managers are implemented by organisations.

3.8. Limitations

Some limitations presented themselves while collecting the data.

During the literature review, many technologies and their related cybersecurity concerns were named. It is impractical to address each of these items here, but the information and references provided can aid those searching for mitigations to the specific technology concerned.

It is known that some protected areas and reserves make use of wireless Wi-Fi mesh networks (Connected Conservation, 2022), but due to the nature of a VM simulated network, Wi-Fi capabilities were inaccessible.

Drones are also well-known in the anti-poaching space, but in the VM simulation here, a "drone" could only be attacked as if it was a virtual computer, but some drone attacks are GPS based and this was not possible to simulate in the simulated environment.

Some attack mitigations called for the implementations of firewalls, and the option of implementing honeypots, demilitarized zones (DMZs) and sniffers was considered. However, even though GNS3 is free and open source, some of the appliances that can be imported are proprietary and require payment or a license, such as Cisco firewalls. It was also deemed unfeasible to incorporate honeypots, DMZs, and sniffers at this stage, but it is a proposal for future work.

Wireshark would have been an ideal way to monitor network traffic and provide further insight into the various attacks and mitigations, but its use was limited in this regard here. During Wireshark's initialisation, the Kali interface became increasingly slower and just before completion, the entirety of the Kali machine would freeze and not recover. This was repeatedly tested both before the attack was launched and during the attack. It was also noticed that there was an uptick in the local machine's RAM usage, which was already very high, indicating a lack of suitable hardware. The Kali machine had to be shut down in GNS3 and restarted, which results in a new Kali instance and has no remnants of the hping3 attack. Wireshark was tested on the local computer outside of the VM, but it ascribes different IP addresses to the GNS3 components than the VM does, and therefore did not prove useful.

A further limitation was that this simulation is not the full physical network, as it was decided only a sample representation was necessary. The available hardware also did not have the capacity for a more encompassing network.

Authentication and authentication attacks could be further explored, but the scope requires only that it is seen as a risk and does not require in-depth analysis and attacks. While some components in the simulation did require passwords, such as the server, the possibility was explored to attack it, but was deemed unfeasible at that point.

3.9. Ethics

The idea to virtually simulate an anti-poaching system was adopted, as this posed no threat to any existing systems, humans, or animals. Due to the sensitive and classified nature of anti-poaching, it was decided that only publicly available data and research would be reported on. This in turn would hopefully not jeopardise current poaching operations. Therefore, the research, methodology and data presented here should be of no ethical concern, as everything possible was done to maintain objectivity and sensitivity around the subject matter.

Further, the proposal and intentions for this research was ethically cleared by the relevant supervisor and Stellenbosch University's Ethical Clearance Committee.

3.10. Conclusion

Even with the above limitations, it was deemed that sufficient data was gathered regarding inherent cyber risks within the hypothetical anti-poaching system. All the attacks were relatively simple in nature and easy to deploy with publicly available tools. Authentication has inherent security risks, but there exist ways in which this can be ameliorated. This shows that there is serious risk in an open and unconfigured network, utilising basic authentication mechanisms. The mitigations for the cyber-attacks required more understanding of the switch and its console to implement, but where possible, it could be seen that the mitigations could protect the network.

4. Findings

4.1. Introduction

The reader should now have a good understanding of an anti-poaching system and the various threats that it can face. This chapter's purpose is to summarise, table, diagram and expand on the various findings made up to this point. Tables and visual depictions will aid the reader in gaining a better perception of the threats and how they affect an anti-poaching system. The reader will be reminded of some key points explained in Chapter 2. The reader will be able to view a summary of the various cyber-attacks performed in Chapter 3. From this table it can be determined if the anti-poaching network simulation and cyber-attack exercise was successful. The threat modelling performed will be depicted in attack trees and tabled in a composite STRIDE/DREAD threat model, providing a holistic overview of the class and risk posed by all the analysed threats. A diagram is then provided on the most common network threats and their mitigations. The findings on authentication will also be presented.

4.2. Literature Lessons

From the literature review that was done in Chapter 2, some key learnings stood out and are noted below.

Enough information is publicly available with which one can infer the workings of an anti-poaching system. Academic literature, governmental guidelines and private initiatives all contribute to this body of knowledge.

Technology is advancing in the anti-poaching arena but may also advance for the poachers. Technology is becoming more ingrained in the daily operations of anti-poaching units and must be adequately secured to protect animal and human lives.

The identified technologies are all host to their own specific cybersecurity vulnerabilities, and nothing is secure by default. Appropriate steps need to be taken to secure the various devices, as well as educate the people involved with these technologies.

It was established that cybersecurity revolves around risk, and that understanding and calculating risk is one way in which the security of assets and a business' cybersecurity goals can be prioritised.

Risk calculations are not easily done with the wide diversity of threats out there. Many threats exist, and this study could only accommodate the analysis of a limited number of the most prevalent of them. Threat modelling was considered to attempt risk identification, calculation, and classification with.

Authentication was touched on as the entry point past a network's defences and it was shown that improvements need to be made to a typical password-based system.

4.3. Data Gathered

A successful hypothetical anti-poaching network simulation was created in GNS3 on a Windows machine, which could simulate a Kali Linux device with which cyber-attacks could be performed. The simulation was a small virtual representation of a physically larger hypothetical anti-poaching system. A larger simulation was unfeasible on the given hardware, but the simulation provided the opportunity to attack a variety of network components. The attacks performed generally disrupted the smooth operation of the network and the various components' ability to communicate with the wider network.

4.3.1. Cyber-Attacks and Mitigations Summary

Table 14 lists the attacks that have been executed on the network, the steps taken to counter them and what the result of the exercise was. The type of attack is also indicated, with an attack being classified as either MitM, DoS or Layer 2 (L2) attack. Port Scanning was classified as a reconnaissance (Recon) effort.

Table 14: Cyber-attacks summary

Type	Attack	Effect on Network	Mitigation	Result
MitM	ARP Poisoning	Pings were unsuccessful and suffered timeouts	DHCP snooping and ARP inspection applied to switch	Switch not reachable, but alerts appear indicating mitigations
MitM	DHCP Spoofing	No effect, failed attack.	DHCP snooping (not applied here for this attack)	N/A
DoS	DHCP Starvation	Pings suffered higher latency or timeouts	DHCP snooping and rate-limiting applied to switch	Enabled the switch to limit and filter traffic

				coming from the Kali machine
MitM	STP Attack	The Kali machine assumed the root role of the switch	portfast, bpduguard and rootguard applied to switch	Ensured continued authority of the switch and disabled the Kali machine's ability to interfere
DoS	TCP/SYN Flood	Pings were intermittently successful, but suffered timeouts and high latency	Firewall (Not applied here)	N/A
DoS	CAM Overflow	Pings suffered occasional timeouts and higher latency	Activated port security on the Kali machine's port	Pings were successful with normal latency
L2	VLAN Hopping	Kali machine creates a trunk link and can access network traffic	Disabled DTP and trunking	No trunking was detected
Recon	Port Scanning	Successful, but no effect. Attacker can use information gained to deploy further attacks	Firewall (not applied here)	N/A

Disregarding the instances of failed attacks and mitigations, the above table clearly shows that several attacks can affect the hypothetical network, and that there is evident risk involved in operating an anti-poaching system. However, the table also shows that these attacks can be prevented with appropriate network configurations and component implementations.

4.3.2. Threat Model

This section will combine the data gathered surrounding threats, attacks, and threat models into a summary analysis.

Figure 39 depicts a high-level overview of a classic attack tree, where the root node is the goal (Poach Animal), and the branching leaf nodes are all different methods with which to achieve this goal. The tree shows methods, such as social engineering, but this is to demonstrate how an attack tree can be expanded upon, where different attack sources are incorporated. The focus here will be placed on the green block, on the branches below "Breach JOC". This branch will be drilled down into in Figure 40.

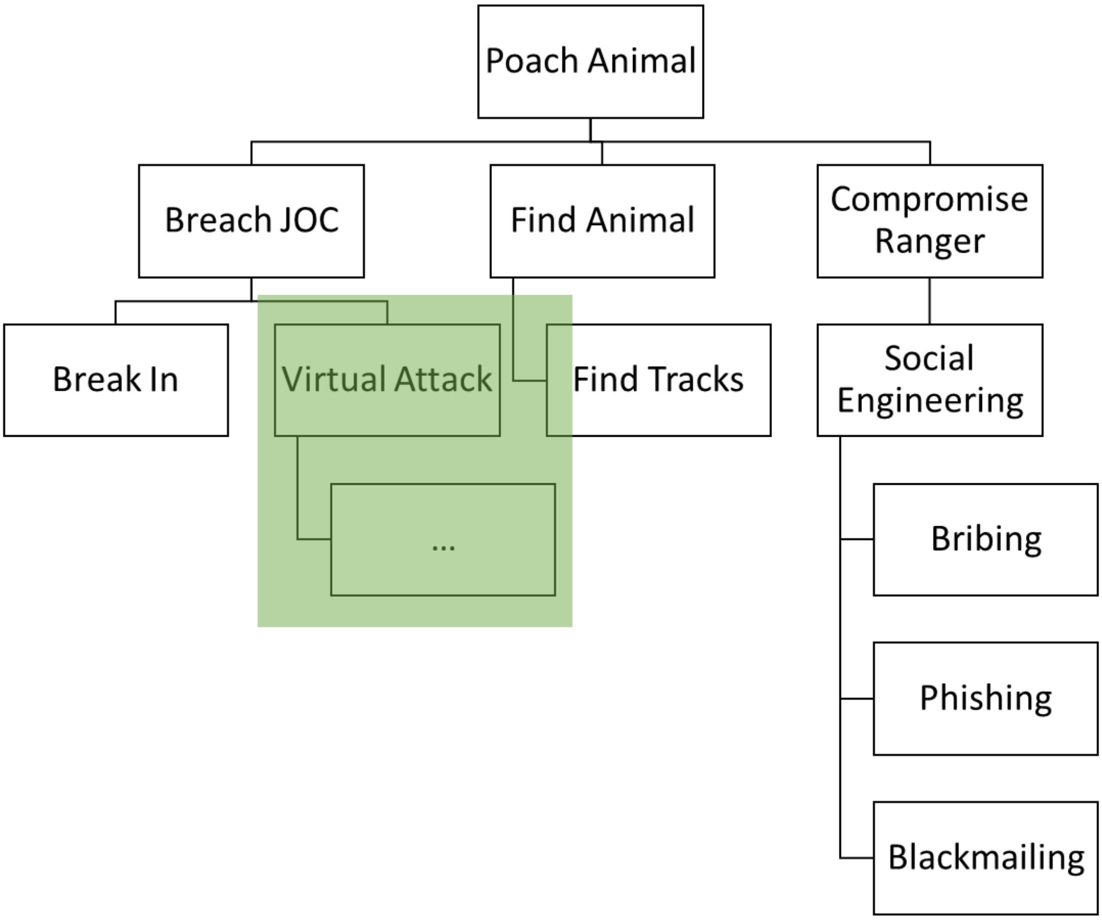


Figure 39: Higher level "Poach Animal" attack tree

Figure 40 shows a drilled down and more detailed attack tree for the "Breach JOC" branch. Due to space limitations, and the specific design of this attack tree, it was decided to show only the specific cyber-attacks performed in Chapter 3. This was done to contextualise how these attacks can come into play in an attack scenario, or how an attacker would come to consider them. An exhaustive attack tree would exceed the current scope requirements and is therefore not modelled here. Subsequent attack trees will take the other discussed threats into account. It was considered to add possible or impossible descriptions to each node, but was deemed unnecessary, as all paths, except the greyed out "Break In", are possible.

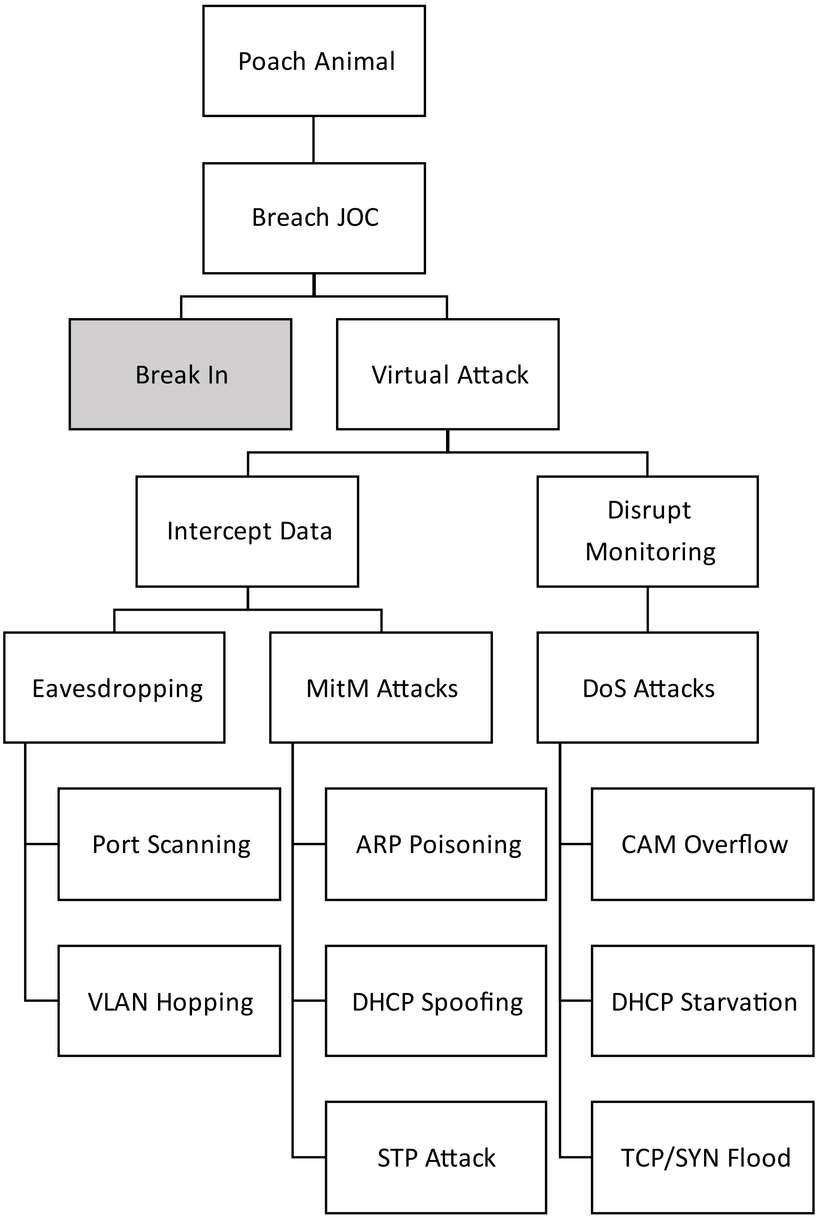


Figure 40: Drilled down "Poach Animal" attack tree

Figure 41 illustrates a tree diagram that does not have a specific attacking goal as the root node, but rather depicts three threat sources and what threats originate from them in practice. This tree diagram is based on Whitman's (2004) research. Each node can be further specified, per Whitman's (2004) findings, but this diagram is only intended as an illustration of the use of attack trees to pinpoint major threats sources and provide an overview relevant to this study.

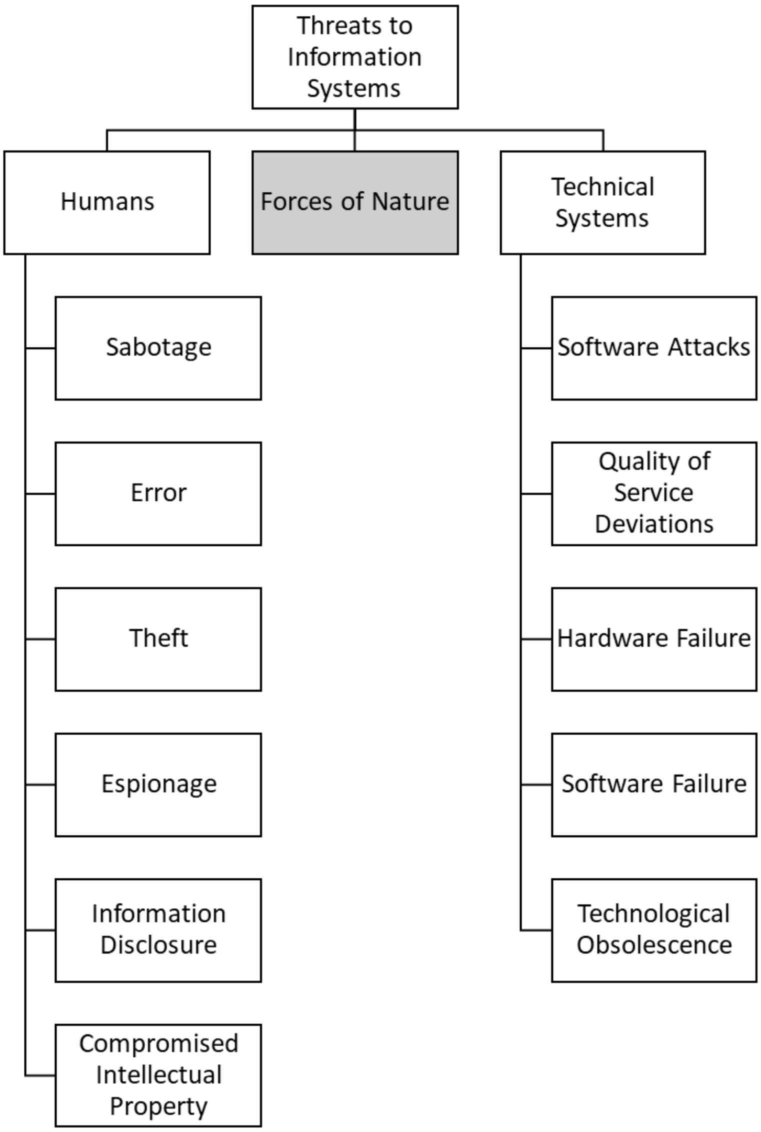


Figure 41: Threats to information systems tree based on Whitman (2004)

Table 15 shows the combined STRIDE and new DREAD model's classifications and scoring, for threats (cyber-attacks) investigated in this study and general common threats identified by Borges (2021). The threats' scores were compared to findings made by Naagas and Palaoag (2018), but are also informed by Omotosho, Haruna and Olaniyi (2019), and Zhang, et al.'s (2021) research.

Table 15: STRIDE/DREAD threat and risk score table

Threat	STRIDE	D	R	E	A	D	Risk	Level
Malware								*High*
-Viruses	T	10	10	10	10	10	10	High
-Rogue Software	S	10	10	5	5	5	7	High
-Trojans	S	10	10	5	5	5	7	High
-Spyware	I	10	10	5	5	5	7	High
-Worms	T	10	10	5	5	5	7	High
DoS Attacks								*High*
-DHCP Starvation	D	10	10	5	10	10	9	High
-TCP/SYN Flood	D	10	10	5	10	10	9	High
-CAM Overflow	D	10	10	5	10	10	9	High
MitM Attacks								*High*
-ARP Poisoning	S	10	10	5	5	5	7	High
-DHCP Spoofing	S	10	10	5	5	5	7	High
-STP Attack	E	10	10	5	10	5	8	High
Phishing	I	10	10	5	10	5	8	High
Rootkits	E	10	10	10	10	10	10	High
SQL Injections	T	10	10	5	10	5	8	High
VLAN Hopping	I	10	10	5	10	5	8	High

Table 15 shows that cyber-attacks and threats pose a significant risk to a network and information system, with threats consistently scoring high across the board. In all cases the damage potential (D) and reproducibility(R) of the threat were at a maximum, indicating that it is quite likely for these threats to surface and for them to wreak havoc on the network. It is important to remember that these threats were all found and measured against an unprotected system, to aid in firstly identifying threats, and then to formulate appropriate mitigations.

While viruses and rootkits both have the highest risk scores, they are different classes of threat. When taking STRIDE categories into account, Table 16 shows one that the D class (Denial of Service), and the E (Elevation of Privilege) class has the highest average risk score of all the classes, with a 9 each.

Table 16: STRIDE risk scores and levels

STRIDE	**Total Risk Score**	**Number of Threats**	**Average Risk Score**	**Average Risk Level**
S	28	4	7	High
T	25	3	8.33	High
R	-	-	-	-
I	23	3	7.67	High
D	27	3	9	High
E	18	2	9	High

This information can be combined with an attack tree, demonstrating the threats as they pertain to STRIDE and their DREAD-derived risk level. This offers a visual depictions of the problematic threats to a network and can be seen in Figure 42. This study was unable to simulate and therefore cover cyber-attacks for the STRIDE class of “Repudiation”. Shostack (2014) was consulted for applicable Repudiation threats and inserted into the tree diagram. This shows a more completed overview of STRIDE threats, but the “R” branch was left greyed out, with no risk estimation, to acknowledge the lack of data for this class.

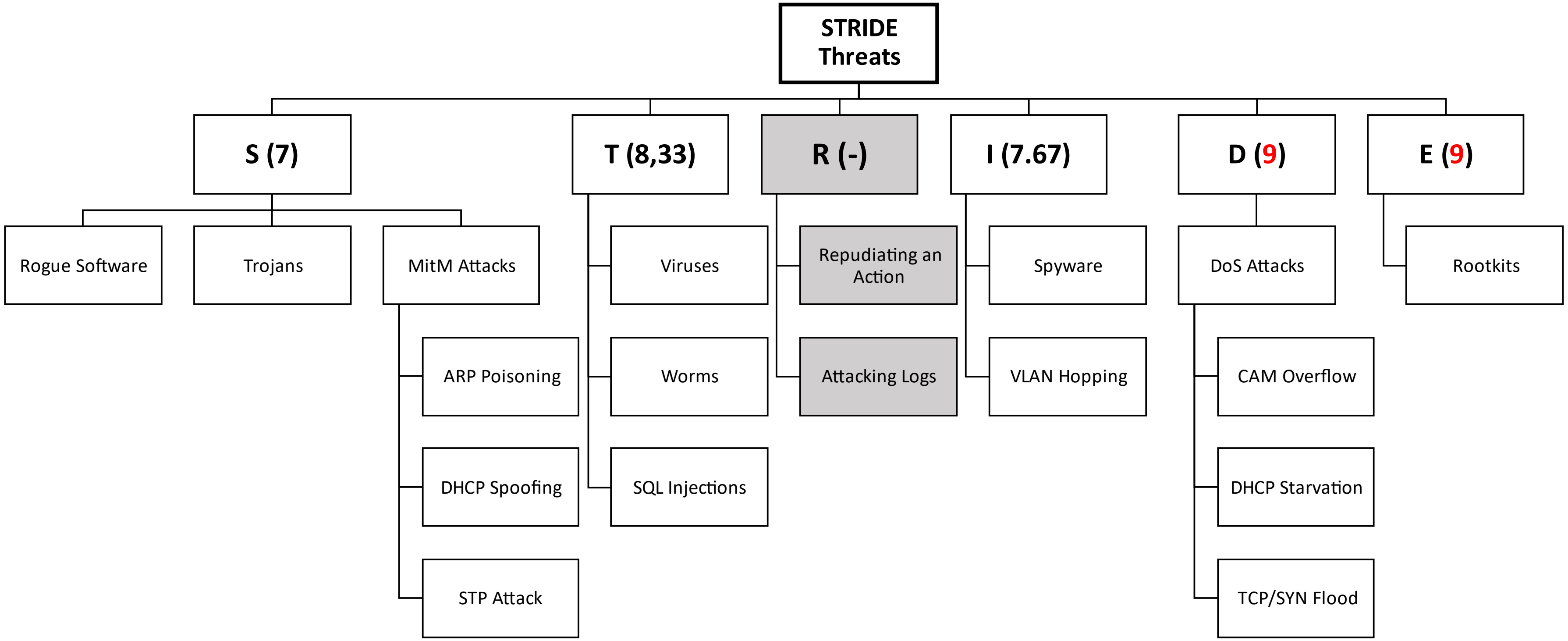


Figure 42: STRIDE threat tree diagram with risk scores per threat class

Attack trees usually do not show which mitigations need to take effect for each attack or threat, but a tree diagram is still a good way to represent this. Figure 43 is a tree diagram showcasing the most common identified threats (Borges, 2021) and their mitigations. Many mitigations derive from Naagas and Palaoag (2018). The mitigations of the specific cyber-attacks performed are referred to in Chapter 3. In the case of a threat such as DoS attacks, the general mitigation recommended by Naagas and Palaoag (2018) were listed as a direct mitigation of the threat, while the specific cyber-attack sub-threats' simulated mitigations were recorded. The mitigations for phishing were learned from Tucker (2022), while Maayan (2021) informed the rootkit mitigations.

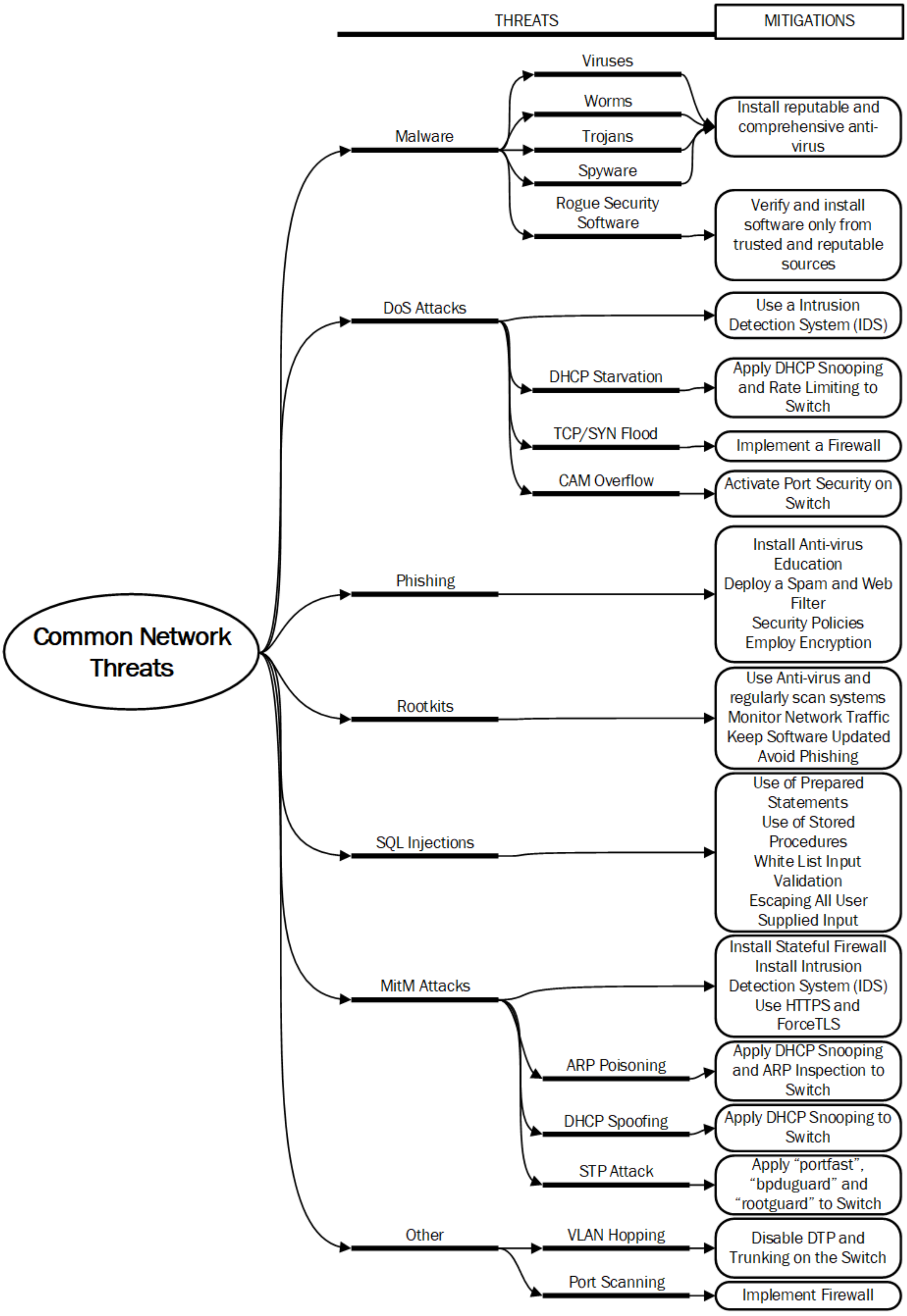

THREATS
MITIGATIONS
Common Network Threats
Malware
Viruses
Worms
Trojans
Spyware
Rogue Security Software
Install reputable and comprehensive anti-virus
Verify and install software only from trusted and reputable sources
DoS Attacks
Use a Intrusion Detection System (IDS)
DHCP Starvation
Apply DHCP Snooping and Rate Limiting to Switch
TCP/SYN Flood
Implement a Firewall
CAM Overflow
Activate Port Security on Switch
Phishing
Install Anti-virus
Education
Deploy a Spam and Web Filter
Security Policies
Employ Encryption
Rootkits
Use Anti-virus and regularly scan systems
Monitor Network Traffic
Keep Software Updated
Avoid Phishing
SQL Injections
Use of Prepared Statements
Use of Stored Procedures
White List Input Validation
Escaping All User Supplied Input
MitM Attacks
Install Stateful Firewall
Install Intrusion Detection System (IDS)
Use HTTPS and ForceTLS
ARP Poisoning
Apply DHCP Snooping and ARP Inspection to Switch
DHCP Spoofing
Apply DHCP Snooping to Switch
STP Attack
Apply "portfast", "bpduguard" and "rootguard" to Switch
Other
VLAN Hopping
Disable DTP and Trunking on the Switch
Port Scanning
Implement Firewall

Figure 43: Common threats and their mitigations

4.3.3. Authentication

It is clear from the literature that while passwords can be at risk, they are a mainstay when it comes to authentication. Therefore, it may be easier to improve upon passwords and their practices than attempt to eliminate them as a form of authentication. A list of do's and do not's, and a list of good recommendations for the formulation of a password were made available by CERN (2018). MFA and password managers stood out as the two main methods with which password insecurity can be combatted.

With regards to common threats and the attacks performed here, it is a necessity to implement more than just passwords on an anti-poaching system. Ali, Dida, and Sam's (2020) research, shown in Table 13, indicated that biometric authentication proved to be a good countermeasure against most threats. Ali, Dida, and Sam (2020) specifically named fingerprint biometrics as the preferred biometric mechanism, also particularly for MitM and DoS attacks.

A lack of awareness, understanding or trust of password managers can result in low adoption rates. If implemented and used correctly, password managers make creating and keeping strong and unique passwords efficient and easy.

Combining passwords and biometrics, especially fingerprints, can constitute a secure MFA mechanism. As anti-poaching systems are highly confidential, it can be considered to add additional authentication factors, and extend the requirements of existing ones.

For the JOC, the geographical location authentication mechanism can be considered, especially for logging on to the system by a user. This will ensure that only those present, and internal personnel can perform certain activities. Table 17 shows how such form of MFA is envisioned in practice, and Figure 44 is a sample scenario for the log on activity of a user. In such a scenario, it is also possible that if certain "suspicious" parameters are specified, a failed login attempt can generate an alert that is sent to the system administrator, as a warning.

Table 17: Example of MFA implemented for the JOC

	Authentication Required By:		
Personnel Activity	**Password**	**Fingerprint**	**Location**
Enters JOC		X	X
Logs onto JOC Computer	X	X	X
Logs onto Laptop	X	X	
Accesses CMORE	X	X	
Accesses Sensitive Information	X	X	
Decrypts data	X	X	
Changes Password	X	X	
System Administrator Functions	X	X	
Logs off JOC Computer/Laptop			
Exits JOC		X	

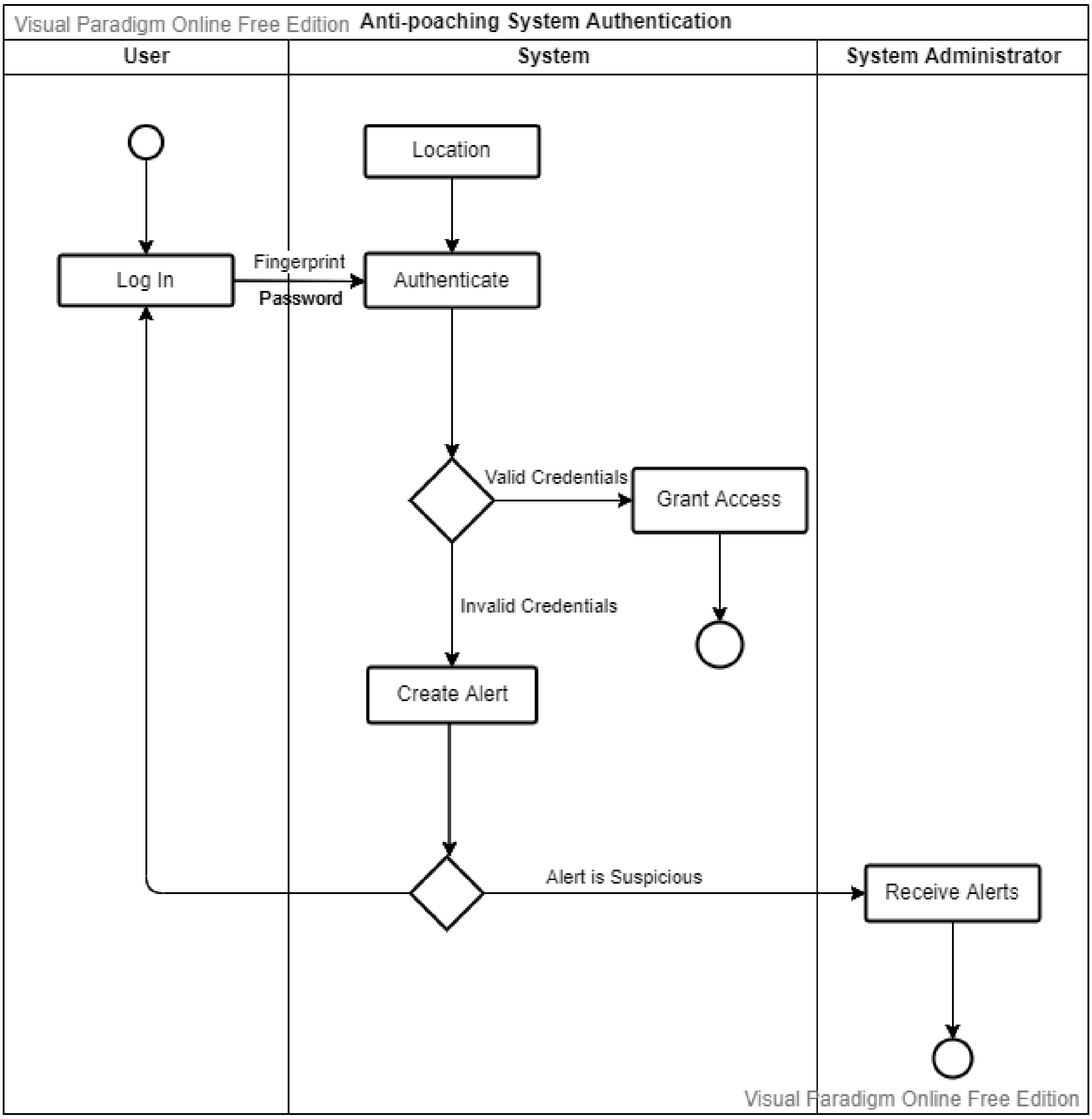


Figure 44: Activity diagram for a user logging onto the system

For rangers in the field, the MFA combination of passwords and fingerprints can also be highly secure. Rangers may need to log onto mobile phones (typically able to process biometrics) and be able to access certain systems, such as CMORE. Most smartphones require a backup PIN or pattern in case the biometrics are not recognized, adding an additional layer of security.

4.4. Conclusion

The literature and gathered data have now been combined in the form of comprehensive results, or findings. The research for this study is now complete. This chapter’s contents will not be summarised here, as the next chapter will briefly point out the main findings and investigate whether they are able to answer the proposed research questions.

5. Conclusion

5.1. Summary of Findings

A hypothetical anti-poaching system can be simulated and attacked in various ways, mostly via MitM and DoS attacks. However, mitigations can also be successfully applied.

With the available threat data, gleaned from literature and virtual experimentation, threat modelling took place successfully. Attack trees captured visual depictions of threat sources and scenarios, while the STRIDE and DREAD models were used to classify and calculate risk. Individually, it appears viruses and rootkits pose significant threats to a network, but as a STRIDE class, Denial of Service (D) and Elevation of Privilege (E) maintained the highest average risk levels.

The most common network threats were focussed on during this study, and this culminated in the creation of a comprehensive Threat-Mitigation tree diagram (Figure 43). This diagram lists the prevalent threats, with added threats and sub-threats in the form of the cyber-attacks performed. Research and virtual experimentation informed all the recorded mitigations.

If authentication is implemented as securely as possible, with password managers and MFA, it lessens the risk for certain cyber-attacks to gain a foothold on the network, as certain authentication mechanisms have been proven to remain resilient against attacks such as MitM and DoS.

5.2. Research Answers

The research questions will now be revisited to see if the data and findings support answers to the three questions.

***RQ1**: Can cybersecurity risks be adequately identified within anti-poaching systems and mitigated using simulation tools?*

Yes. Risks were adequately identified with literature, practical experimentation with a simulation tool and threat modelling.

There is a vast quantity of academic resources available on information system risks and threats, as well as cybersecurity countermeasures. The literature provided a variety of risks for consideration, and these were incorporated into a simulation and a threat model.

Cyber-attacks were successfully performed in the simulation, proving that there is inherent risk in an unconfigured anti-poaching network system. However, only eight cyber-attacks were performed on the simulated network, mostly DoS and MitM attacks, as other types of attacks were not always possible within the simulation, such as password cracking or deploying a virus. The simulation also provided most of the capabilities with which one could mitigate these eight attacks.

The literature on threat models further aided in identifying the risks associated with information systems. The threat models classified and calculated risks.

***RQ2:** Can the risks identified in an anti-poaching system be calculated with threat models?*

Yes. Two threat models were employed, STRIDE and DREAD. A modified version of the traditional DREAD model proved useful in quantitatively scoring and calculating the risk of a threat, and then translating it into qualitative terms based on the average risk score. Classifying a risk with the STRIDE threat model and scoring it with the improved DREAD model ensures one can prioritise risks and see which classes of risks are the most problematic for a system.

***SQ1**: What cybersecurity countermeasures are currently in place to protect the anti-poaching industry and what additional countermeasures can be added or modified to improve the safety of animals?*

This question cannot be answered with the available data. While sources exist on anti-poaching systems in some form, no sources were found on what cybersecurity countermeasures are currently in place on these systems. Further, no person working with such a system was found to be available for comment, even though contact was made with several organisations actively involved in this. Therefore, additions and modifications cannot be made in confidence. Recommendations can be made, but will assume an unprotected system, and these recommendations will be general in nature or specific to the risks identified in this study.

5.3. Recommendations

The Threat-Mitigation diagram in Figure 43 provides a good overview of what should be implemented on a network to avoid the most common threats as a baseline. Briefly, the key recommendations that stand out from this study as necessities when setting up an anti-poaching system, are the following:

- Install a comprehensive and reputable antivirus
- Keep software up to date
- Use Intrusion Detection Systems (IDS)
- Implement a firewall with appropriate configurations
- Educate personnel and create security awareness

And the most important thing is to set up the system as secure as possible:

- Have an expertly and securely configured network, router, and switch configuration.

These are by no means an exhaustive list and on its own does not guarantee complete and total security. Therefore, it is recommended that at least one dedicated and well-rounded cybersecurity expert is employed in an anti-poaching operation, that can manage current system risks and adapt the system when and where necessary, as risks evolve. It can also be investigated whether it may be feasible and beneficial for an independent audit of cybersecurity to be performed at least annually.

While not performed here, regular, and thorough risk analyses and risk assessments can complement threat modelling and be compiled into a risk framework document to guide the cybersecurity operations of the anti-poaching units.

With regards to authentication, MFA is considered a must, where the combination of passwords and fingerprint biometrics are recommended. For certain use cases, geographical location factors can be incorporated. Passwords should be strong and unique, and each organisation can customise their password guidelines according to best security practices. The use of an enterprise password manager should also be considered, as this can safely generate and store highly secure passwords, avoiding the risk of insecure password behaviour.

5.4. Future Research

During the formulation and execution of this study, it became apparent that there were unavoidable research gaps. These gaps arise from the lack of literature or information, inadequate resources and from scope limitations.

The hypothetical network simulation provided a good starting point for risk analysis, but to accurately test the robustness of an anti-poaching system, an existing physical system would need to be assessed. This may also provide the opportunity to assess components that were limited in the simulation. Tests could be performed on radio masts, Wi-Fi implementations and drones, if applicable. More sophisticated cybersecurity tools can be introduced to protect the system and the effects of honeypots, DMZs, firewalls, and sniffers could be observed on the network.

A practical way of gaining knowledge of an existing anti-poaching system would be to gather information via surveys, interviews, and observations. The problem with this is that anti-poaching organisations are highly unlikely to divulge such sensitive and current information. They may allow a highly secure risk assessment by an expert but will likely disallow its publication.

This study is concerned with technical risks related to an information system. As previously mentioned, risk comes in many forms and an extensive and comprehensive risk assessment can be done on all risk aspects relevant to an anti-poaching system. The full range of technical risks alone can also inform a future study, where more cyber-attacks are performed on the network.

There exists a multitude of threat modelling techniques and it can be investigated whether there is a more suitable model to apply to an anti-poaching system, depending on newly gathered information from future work. The traditional and modified DREAD models can be further compared and evaluated for accuracy. Standardising and improving a form of the DREAD threat model can make it easier to apply to different use cases with a high degree of accuracy.

Threat modelling and risk calculations based on cyber-attacks were focused on, but one can also perform the steps of risk analyses and risk assessments to develop a complete risk framework for anti-poaching systems.

While the authentication of people on systems was reviewed, further research can be done into creating a more robust system of authentication, via access control, MFA, and cryptography, that is practical for an anti-poaching system.

6. References

Abdulhafiz, N., Faith, E. & Oyenike, O., 2020. Mitigating DHCP Starvation Attack Using Snooping Techniques. *FUDMA Journal of Sciences (FJS)*, 4(1), pp. 560-566.

Abomhara, M., Gerdes, M. & Køien, G. M., 2015. A STRIDE-Based Threat Model for Telehealth Systems. *Norsk informasjonssikkerhetskonferanse (NISK2015)*, pp. 1-15.

Afribary, 2021. *What is the Difference Between the Theoretical and the Conceptual Framework?*. [Online] https://afribary.com/blog/5/what-is-the-difference-between-the-theoretical-and-the-conceptual-framework/#:~:text=What%20is%20the%20difference%20between%20the%20conceptual%20and%20the%20theoretical,will%20have%20to%20be%20explored.&text=The%20theoretical%2 [Accessed 27 January 2022].

Aldaoud, M., Al-Abri, D., Maashri, A. A. & Kausar, F., 2021. DHCP attacking tools: an analysis. *Journal of Computer Virology and Hacking Techniques*, Volume 17, pp. 119-129.

Ali, G., Dida, M. & Sam, A., 2020. Two-Factor Authentication Scheme for Mobile Money: A Review of Threat Models and Countermeasures. *Future Internet*, 12(160), pp. 1-27.

Almehmadi, T. & Alsolami, F., 2019. Password Security in Organizations: User Attitudes and Behaviors Regarding Password Strength. *16th International Conference on Information Technology - New Generations (ITNG)*, Volume 800, pp. 9-13.

Amin, Z., 2017. A practical road map for assessing cyber risk. *Journal of Risk Research*, pp. 1-12.

Anwar, M. N., Nazir, M. & Ansari, A. M., 2020. Modeling Security Threats for Smart Cities: A STRIDE-Based Approach. In: S. Ahmed, S. Abbas & H. Zia, eds. *Smart Cities—Opportunities and Challenges. Lecture Notes in Civil Engineering, vol 58.* Singapore: Springer, pp. 387-396.

Ball, M. B., Wenham, C. M., Clegg, B. & Clegg, S. B., 2018. What does it take to curtail rhino poaching? Lessons learned from twenty years of experience at Malilangwe Wildlife Reserve, Zimbabwe. *Pachyderm,* Volume 60, pp. 96-104.

Barnett, P., 2019. *Guide: How to Block or Allow TCP/IP Port in Windows Firewall.* [Online] https://www.action1.com/how-to-block-or-allow-tcp-ip-port-in-windows-firewall/ [Accessed 19 May 2022].

Bergs, C.-J., Bruiners, J., Fakier, F. & Stofile, L., 2021. Cyber Security and Wind Energy: A Fault-Tolerance Analysis of DDoS Attacks. *International Conference on Cyber Warfare and Security,* pp. 1-13.

Bhushan, B. & Sahoo, G., 2017. Recent Advances in Attacks, Technical Challenges, Vulnerabilities and Their Countermeasures in Wireless Sensor Networks. *Wirelesss Personal Communications,* Volume 98, pp. 2037-2077.

Bhushan, B., Sahoo, G & Rai, A. K., 2017. Man-in-the-middle attack in wireless and computer networking — A review. *2017 3rd International Conference on Advances in Computing,Communication & Automation (ICACCA),* pp. 1-6.

Borges, E., 2021. *Top 10 Common Network Security Threats Explained.* [Online] https://securitytrails.com/blog/top-10-common-network-security-threats-explained [Accessed 30 June 2022].

Bridge, A. S. et al., 2019. An Arduino-Based RFID Platform for Animal Research. *Frontiers in Ecology and Evolution,* 7(257), pp. 1-10.

Burrows, T., 2022. *Organisational culture as the last line of defence in cyber security.* [Online] www.itweb.co.za/content/8OKdWqDXN5QqbznQ#Echobox=1652789607 [Accessed 31 May 2022].

BusinessWire, 2018. *Dimension Data and Cisco Take Anti-Poaching Technology into Africa.* [Online] https://www.businesswire.com/news/home/20180508006383/en/Dimension-Data-and-Cisco-Take-Anti-Poaching-Technology-into-Africa [Accessed 30 January 2022].

Cellan-Jones, R., 2011. *A 15 pound computer to inspire young programmers.* [Online] https://www.bbc.co.uk/blogs/thereporters/rorycellanjones/2011/05/a_15_computer_to_inspire_young.html [Accessed 13 June 2022].

CERN, 2018. *Password Recommendations.* [Online] https://security.web.cern.ch/recommendations/en/passwords.shtml [Accessed 28 July 2022].

Cesar, P. & Pinter, R., 2019. Some ethical hacking possibilities in Kali Linux environment. *Journal of Applied Technical and Educational Sciences*, 9(4), pp. 129-149.

Chapman, L. A. & White, P. C. L., 2019. Anti-poaching strategies employed by private rhino owners in South Africa. *Pachyderm,* Volume 61, pp. 179-183.

Ciampa, M., 2015. *Security+ Guide To Network Security Fundamentals.* Boston: Cengage Learning.

Cisco, 2022. *IOSvL2.* [Online] https://developer.cisco.com/docs/modeling-labs/#!iosvl2/iosvl2 [Accessed 8 July 2022].

Cisco, n.d.. *Configuring DHCP Snooping.* [Online] https://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst6500/ios/12-2SXF/native/configuration/guide/swcg/snoodhcp.pdf
[Accessed 17 May 2022].

Cloudflare, 2022. *What is a router?.* [Online] https://www.cloudflare.com/learning/network-layer/what-is-a-router/ [Accessed 6 June 2022].

Connected Conservation, 2022. *Technology Solutions.* [Online] https://connectedconservation.foundation/technology/ [Accessed 30 January 2022].

Cooney, R. et al., 2016. From Poachers to Protectors: Engaging Local Communities in Solutions to Illegal Wildlife Trade. *Conservation Letters*, 10(3), pp. 367-374.

CSIR, 2022. *CMORE.* [Online] https://www.csir.co.za/cmore [Accessed 10 February 2022].

Davis, A., 2021. *Reconnecting with nature in your classroom.* [Online] https://helloworld.raspberrypi.org/articles/HW14-reconnecting-with-nature-in-your-classroom [Accessed 13 June 2022].

Department of Environmental Affairs, 2010. *National Strategy For The Safety And Security Of Rhinoceros Populations In South Africa.* [Online] http://www.stoprhinopoaching.com/wp-content/uploads/2021/08/DEA-National-Rhino-Strategy.pdf [Accessed 3 February 2022].

Department of Environmental Affairs, 2020. *Guidelines to Inform the Establishment of Anti-Poaching Related Systems and Services.* [Online] https://www.environment.gov.za/sites/default/files/legislations/guidelinesforantipoaching_systemsestablishment.pdf [Accessed 16 October 2021].

Dimension Data, 2020. *Connected Conservation Foundation Lauches.* [Online] https://www.itweb.co.za/content/rxP3jMBmB227A2ye [Accessed 6 June 2022].

Duffy, R., 1999. The role and limitations of state coercion: Anti-poaching policies in Zimbabwe. *Journal of Contemporary African Studies,* 17(1), pp. 97-121.

Eloff, C. & Lemieux, A. M., 2014. Rhino Poaching in Kruger National Park, South Africa: Aligning Technology, Analysis and Prevention. In: A. M. Lemieux, ed. *Situational Prevention of Poaching.* London: Routledge, pp. 18-43.

Ettercap, 2022. *Ettercap.* [Online] https://www.ettercap-project.org/# [Accessed 10 May 2022].

Fromaget, P., n.d.. *17 Security Tips To Protect Your Raspberry Pi Like A Pro.* [Online] https://raspberrytips.com/security-tips-raspberry-pi/ [Accessed 13 June 2022].

Garg, S. & Baliyan, N., 2021. Comparative analysis of Android and iOS from security viewpoint. *Computer Science Review,* Volume 40, pp. 1-13.

Gavaskar, S., Surendiran, R. & Ramaraj, E., 2010. Three Counter Defense Mechanism for TCP SYN Flooding Attacks. *International Journal of Computer Applications,* 6(6), pp. 12-15.

GNS3, 2022a. *GNS3.* [Online] https://gns3.com/ [Accessed 5 April 2022].

GNS3, 2022b. *GNS3 Setup wizard with the GNS3 VM.* [Online] https://docs.gns3.com/docs/getting-started/setup-wizard-gns3-vm [Accessed 7 April 2022].

GNS3, 2022c. *GNS3 Windows Install.* [Online] https://docs.gns3.com/docs/getting-started/installation/windows/ [Accessed 7 April 2022].

GNS3, 2022d. *The NAT Node.* [Online] https://docs.gns3.com/docs/using-gns3/advanced/the-nat-node/ [Accessed 10 June 2022].

Gonzalez, C., 2022. *Top 8 Threat Modeling Methodologies and Techniques.* [Online] https://www.exabeam.com/information-security/threat-modeling/#:~:text=There%20are%20six%20main%20methodologies,threats%20facing%20your%20IT%20assets. [Accessed 10 June 2022].

Gope, P. & Sikdar, B., 2019. Lightweight and Privacy-Preserving Two-Factor Authentication Scheme for IoT Devices. *IEEE Internet of Things*, 6(1), pp. 580-589.

Guo, R., 2019. Survey on WiFi infrastructure attacks. *International Journal of Wireless and Mobile Computing,* 16(2), pp. 97-101.

Habib, H. et al., 2018. User Behaviors and Attitudes Under Password Expiration Policies. *Proceedings of the Fourteenth Symposium on Usable Privacy and Security*, pp. 13-30.

Heintzkill, R., 2021. *CAM Table Overflow Attack Explained.* [Online] https://www.cbtnuggets.com/blog/technology/networking/cam-table-overflow-attack-explained [Accessed 18 May 2022].

Hossain, A. et al., 2016. Assessing the efficacy of camera trapping as a tool for increasing detection rates of wildlife crime in tropical protected areas. *Biological Conservation,* Volume 201, pp. 314-319.

Ingalls, S., 2021. *Network Server.* [Online] https://www.serverwatch.com/servers/network-server/ [Accessed 10 June 2022].

International Telecommunications Union, 2022. *Definition of Cybersecurity*. [Online] https://www.itu.int/en/ITU-T/studygroups/com17/Pages/cybersecurity.aspx [Accessed 27 January 2022].

Ion, I., Reeder, R. & Consolvo, S., 2015. "...no one can hack my mind": Comparing Expert and Non-Expert Security Practices. *Symposium on Usable Privacy and Security (SOUPS),* pp. 327-346.

Jana, I., 2016. Effect of ARP poisoning attacks on modern operating systems. *Information Security Journal: A Global Perspective*, pp. 1-6.

Kali, 2021a. *hping3*. [Online] https://www.kali.org/tools/hping3/ [Accessed 10 May 2022].

Kali, 2021b. *Enabling Root.* [Online] https://www.kali.org/docs/general-use/enabling-root/ [Accessed 7 June 2022].

Kali, 2022. *Yersinia.* [Online] https://www.kali.org/tools/yersinia/#:~:text=Yersinia%20is%20a%20framework%20for,the%20deployed%20networks%20and%20systems. [Accessed 10 May 2022].

Kaplan, S. & Garrick, B. J., 1981. On the Quantitive Definition of Risk. *Risk Analysis,* 1(1), pp. 11-27.

Khan, S., Mast, N., Loo, K.-K. & Salahuddin, A., 2008. Passive Security Threats and Consequences in IEEE 802.11 Wireless Mesh Networks. *International Journal of Digital Content Technology and its Application,* 2(3), pp. 4-8.

Lancaster, C., 2018. *How Technology Is Helping Fight Poaching in Kenya.* [Online] https://theculturetrip.com/africa/kenya/articles/how-technology-is-helping-fight-poaching-in-kenya/[Accessed 8 April 2022].

Logix Consulting, 2019. *What Is the DREAD Cybersecurity Model?.* [Online] https://logixconsulting.com/2019/12/18/what-is-the-dread-cybersecurity-model/ [Accessed 14 June 2022].

Maayan, G., 2021. *How to prevent a rootkit attack.* [Online] https://blog.malwarebytes.com/how-tos-2/2020/01/how-to-prevent-a-rootkit-attack/#:~:text=To%20fully%20protect%20yourself%20against,then%20reinstall%20the%20entire%20system.&text=Phishing%20is%20a%20type%20of,or%20downloading%20an%20infected%20attachment.
[Accessed 6 July 2022].

Mahmood, S., Mohsin, S. M. & Akber, S., 2020. Network Security Issues of Data Link Layer: An Overview. *3rd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET),* pp. 1-7.

Mangut, H. A., Al-Nemrat, A., Benzaid, C. & Tawil, A.-r. H., 2015. ARP Cache Poisoning Mitigation and Forensics Investigation. *2015 IEEE Trustcom/BigDataSE/ISPA,* pp. 1392-1397.

Martin, E. D., Kargaard, J. & Sutherland, I., 2019. Raspberry Pi Malware: An analysis of cyberattacks towards IoT devices. *The 10h IEEE International Conference on Dependable Systems, Services and Technologies,* pp. 2-7.

Massawe, E. A. K. M., Kaijage, S. & Seshaiyer, P., 2017. An Intelligent Real-Time Wireless Sensor Network Tracking System For Monitoring Rhinos ad Elephants in Tannzania National Parks: A Review. *International Journal of Advanced Smart Sensor Network Systems,* 7(4), pp. 1-11.

Meek, P. D. et al., 2018. Camera trap theft and vandalism: occurrence, cost, prevention and implications for wildlife research and management. *Remote Sensing in Ecology and Conservation,* pp. 1-9.

Microsoft, n.d.. *How to prevent and remove viruses and other malware.* [Online] https://support.microsoft.com/en-us/topic/how-to-prevent-and-remove-viruses-and-other-malware-53dc9904-0baf-5150-6e9a-e6a8d6fa0cb5 [Accessed 29 June 2022].

Morrow, T., 2018. *12 Risks, Threats, & Vulnerabilities in Moving to the Cloud.* [Online] https://insights.sei.cmu.edu/blog/12-risks-threats-vulnerabilities-in-moving-to-the-cloud/ [Accessed 15 February 2022].

Mukwazvure, A. & Magadza, T. B. H. T., 2014. A Survey on Anti-Poaching Strategies. *International Journal of Science and Research,* 3(6), pp. 1064-1066.

Mulero-Pazmany, M. et al., 2014. Remotely Piloted Aircraft Systems as a Rhinoceros AntiPoaching Tool in Africa. *PLoS ONE,* 9(1), pp. 1-10.

Mutchler, P. et al., 2015. A Large-Scale Study of Mobile Web App Security. *Mobile Security Technologies Workshop (MoST),* pp. 1-11.

Naagas, M. A. & Palaoag, T. D., 2018. A Threat-Driven Approach to Modeling a Campus Network Security. *ICCBN 2018: Proceedings of the 6th International Conference on Communications and Broadband Networking,* pp. 6-12.

Nautiyal, L., Malik, P. & Agarwal, A., 2018. Cybersecurity System: An Essential Pillar of Smart Cities. In: Z. Mahmood, ed. *Smart Cities: Development and Governance Frameworks.* Cham: Springer, pp. 25-50.

Neumann, J. C., 2015. *The Book of GNS3: Build Virtual Network Labs using Cisco, Juniper, and more.* San Francisco: No Starch Press.

Nicholas, P. J. & Alderson, D. L., 2012. Fast, Effective Transmitter Placement in Wireless Mesh Networks. *Military Operations Research,* 17(4), pp. 69-84.

Nmap, 2022. *Nmap.* [Online] https://nmap.org/ [Accessed 3 June 2022].

O'Donoghue, P. & Rutz, P., 2016. Real-time anti-poaching tags could help prevent imminent species extinctions. *Journal of Applied Ecology,* Volume 53, pp. 5-10.

Offensive Security, 2022. *Kali Linux.* [Online] https://www.kali.org/ [Accessed 5 April 2022].

Ometov, A. et al., 2019. Challenges of Multi-Factor Authentication for Securing Advanced IoT (A-IoT) Applications. *IEEE Network,* 3(2), pp. 1-7.

Omotosho, A., Haruna, B. A. & Olaniyi, O. M., 2019. Threat Modeling of Internet of Things Health Devices. *Journal of Applied Security Research,* 14(1), pp. 106-121.

Peltier, T. R., 2010. *Information Security Risk Analysis.* 3rd ed. Boca Raton: CRC Press.

Penny, S. G., White, R. L., Scott, D. M. M. L. & P, P. A., 2019. Using drones and sirens to elicit avoidance behaviour in white rhinoceros as an anti-poaching tactic. *Proceedings of the Royal Society of Britain,* Volume 286, pp. 1-9.

Pevnev, V., Tsuranov, M., Zemlianko, H. & Amelina, O., 2021. Conceptual Model of Information Security. *Integrated Computer Technologies in Mechanical Engineering - 2020 (ICTM 2020) Lecture Notes in Networks and Systems,* Volume 188, pp. 158-168.

Popeskic, V., 2013. *Mitigate VLAN hopping attack – Get rid of Layer 2 attacks.* [Online] https://howdoesinternetwork.com/2012/mitigate-vlan-hopping#:~:text=To%20help%20prevent%20a%20VLAN,for%20the%20native%20VLAN%20assignment. [Accessed 18 May 2022].

Posey, B., 2021. *Risk Management Framework (RMF).* [Online] https://www.techtarget.com/searchcio/definition/Risk-Management-Framework-RMF [Accessed 8 June 2022].

Radhappa, H., Pan, L., Zheng, J. X. & Wen, S., 2018. Practical overview of security issues in wireless sensor network applications. *International Journal of Computers and Applications,* 40(4), pp. 202-213.

Recker, J., 2013. *Scientific Research in Information Systems: A Beginner's Guide.* Heidelberg: Springer.

Reynolds, L., 2022. *Linux IP forwarding – How to Disable/Enable using net.ipv4.ip_forward.* [Online] https://linuxconfig.org/how-to-turn-on-off-ip-forwarding-in-linux [Accessed 18 May 2022].

Rosencrance, L., 2022. *Security Posture.* [Online] https://www.techtarget.com/searchsecurity/definition/security-posture#:~:text=Security%20posture%20refers%20to%20an,to%20ever%2Dchanging%20cyber%20threats. [Accessed 10 June 2022].

Rot, A., 2008. IT Risk Assessment: Quantitative and Qualitative Approach. *Proceedings of the World Congress on Engineering and Computer Science,* pp. 1-6.

Salahdine, F. & Kaabouch, N., 2019. Social Engineering Attacks: A Survey. *Future Internet,* 11(89), pp. 1-17.

Samonas, S. & Coss, D., 2014. The CIA Strikes Back: Redefining Confidentiality, Integrity and Availability in Security. *Journal of Information System Security*, 10(3), pp. 21-45.

Sankar, R., 2015. *macof.* [Online] https://kalilinuxtutorials.com/macof/ [Accessed 7 June 2022].

Save The Rhino, 2020. *Poaching Statistics.* [Online] https://www.savetherhino.org/rhino-info/poaching-stats/ [Accessed 26 January 2022].

SAWC, 2022. *New VHF ear-tagging initiative on rhino begins.* [Online] https://wildlifecollege.org.za/new-vhf-ear-tagging-initiative-on-rhino-begins/ [Accessed 10 June 2022].

Schneier, B., 1999. *Attack Trees.* [Online] https://www.schneier.com/academic/archives/1999/12/attack_trees.html [Accessed 6 June 2022].

Schoenfield, B. S. E., 2015. *Securing Systems: Applied Security Architecture and Threat Models.* Boca Raton: CRC Press.

Shah, M. et al., 2019. Penetration Testing Active Reconnaissance Phase – Optimized Port Scanning With Nmap Tool. *International Conference on Computing, Mathematics and Engineering Technologies – iCoMET*, pp. 1-7.

Shaw, K., 2020. *What is a network switch, and how does it work?.* [Online] https://www.networkworld.com/article/3584876/what-is-a-network-switch-and-how-does-it-work.html [Accessed 10 June 2022].

Shostack, A., 2014. *Threat Modeling: Designing for Security.* Indianapolis: Wiley.

Siddappaji, B., Hajoary, P. K. & Akhilesh, K. B., 2020. Role of Cyber Security in Drone Technology. In: K. B. Akhilesh & D. P. F. Moller, eds. *Smart Technologies: Scope and Application.* Singapore: Springer, pp. 169-178.

Simlai, T., 2015. Conservation 'Wars': Global Rise of Green Militarisation. *Economic and Political Weekly,* 50(50), pp. 39-44.

Singita, 2022. *Anti-poaching Technology.* [Online] https://singita.com/conservation/projects/anti-poaching-technology [Accessed 12 April 2022].

Sotnikov, I., 2022. *How to Perform IT Risk Assessment.* [Online] https://blog.netwrix.com/2018/01/16/how-to-perform-it-risk-assessment/ [Accessed 25 May 2022].

South African National Parks, 2022. *Media Release: Rhino Poaching in South Africa in 2021.* [Online] https://www.sanparks.org/about/news/?id=58507 [Accessed 10 February 2022].

Svensson, E. & Ryden, A., 2019. *JamaicaEye – What Does Cybersecurity Look Like in One Of The Most Recently Developed CCTV Networks?.* [Online] https://www.diva-portal.org/smash/get/diva2:1380862/FULLTEXT01.pdf [Accessed 3 February 2022].

Teofilo, L., n.d.. *Cisco Portfast, BPDU Guard, and Root Guard.* [Online] https://www.flackbox.com/portfast-bpdu-guard-and-root-guard [Accessed 17 May 2022].

Thevar, S. & Bhanot, N., 2021. *How Technology has improved Anti-poaching Efforts.* [Online] https://sanctuarynaturefoundation.org/article/how-technology-has-improved-anti-poaching-efforts [Accessed 8 April 2022].

Trejo, L. A., Monroy, R. & Monsalvo, R. L., 2006. Spanning Tree Protocol and Ethernet PAUSE Frames DDoS Attacks: Their Efficient Mitigation.

Tripathi, N. & Mehtre, B. M., 2014. Analysis of various ARP poisoning mitigation techniques: A comparison. *2014 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT),* pp. 125-132.

Tucker, T., 2022. *Phishing Attack Prevention: How to Identify & Avoid Phishing Scams in 2022.* [Online] https://digitalguardian.com/blog/phishing-attack-prevention-how-identify-avoid-phishing-scams#:~:text=Install%20an%20antivirus%20solution%2C%20schedule,Encrypt%20all%20sensitive%20company%20information. [Accessed 6 July 2022].

VMWare, 2022. *VMWare.* [Online] https://www.vmware.com/ [Accessed 5 April 2022].

Wawrzyniak, D., 2006. Information Security Risk Assessment Model for Risk Management. In: S. Fischer-Hübner, S. Furnell & C. Lambrinoudakis, eds. *Trust and Privacy in Digital Business.* Berlin: Springer.

Welsh, C., 2013. *GNS3 Network Simulation Guide.* Birmingham: Packt Publishing.

Whitman, M. E., 2004. In defense of the realm: understanding the threats to information security. *International Journal of Information Management,* Volume 24, pp. 43-57.

Wireshark, 2022. *Wireshark.* [Online] https://www.wireshark.org/ [Accessed 3 June 2022].

Yaacoub, J.-P., Houra, H., Salman, O. & Chahab, A., 2020. Security analysis of drones systems: Attacks, limitations, and Recommendations. *Internet of Things,* Volume 11, pp. 1-39.

Yahya, A., Bogaisang, K., Gamoshe, O. G. & Maina, M. D., 2019. Anti-poaching System using Wireless Sensors Network. *BIUST Research and Innovation Symposium,* pp. 1-3.

Yang, R., Ford, B., Tambe, M. & Lemieux, A., 2014. Adaptive Resource Allocation for Wildlife Protection agaisnt Illegal Poachers. *Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2014),* pp. 1-8.

Yıldırım, M. & Mackie, I., 2019. Encouraging users to improve password security and memorability. *International Journal of Information Security,* Volume 18, pp. 741-759.

Yue, M., 2015. Security of VHF Data Link in ATM. In: S. M. Musa & Z. Wu, eds. *Aeronautical Telecommunications Network.* 65-92: CRC Press, pp. 65-.

ZAPWing, 2022. *The Rhino Poaching Crisis.* [Online] http://www.zapwing.org/the-rhino-poaching-crisis/ [Accessed 12 April 2022].

Zhang, L. et al., 2021. A risk-level assessment system based on the STRIDE/DREAD model for digital data marketplaces. *International Journal of Information Security,* Volume 21, pp. 509-525.

Appendix A

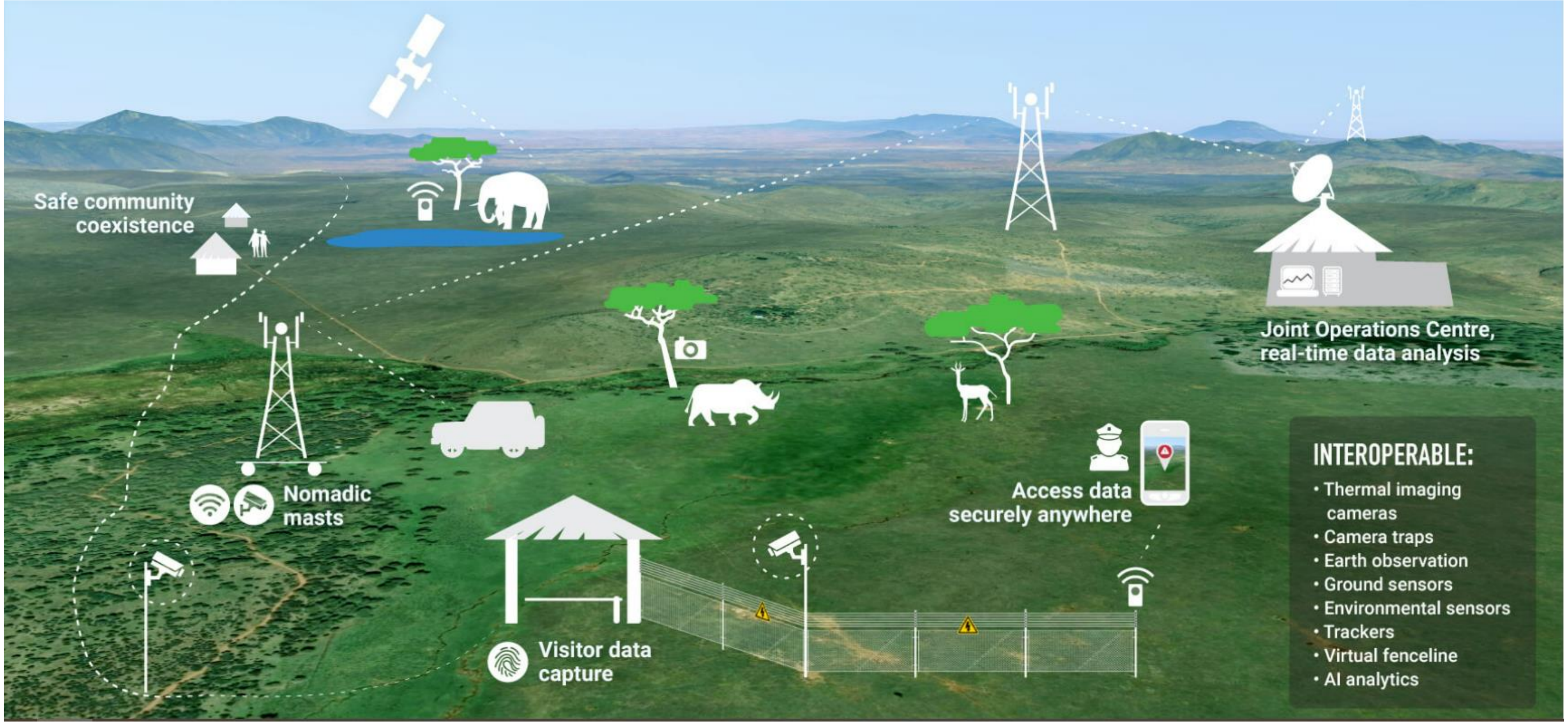
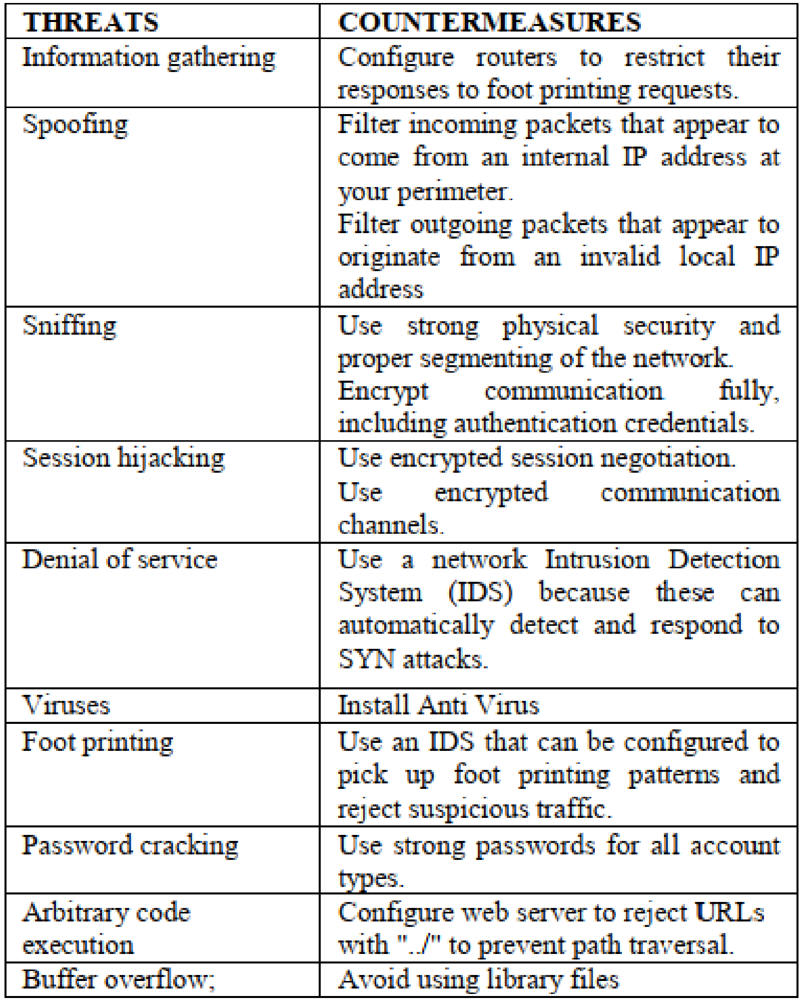
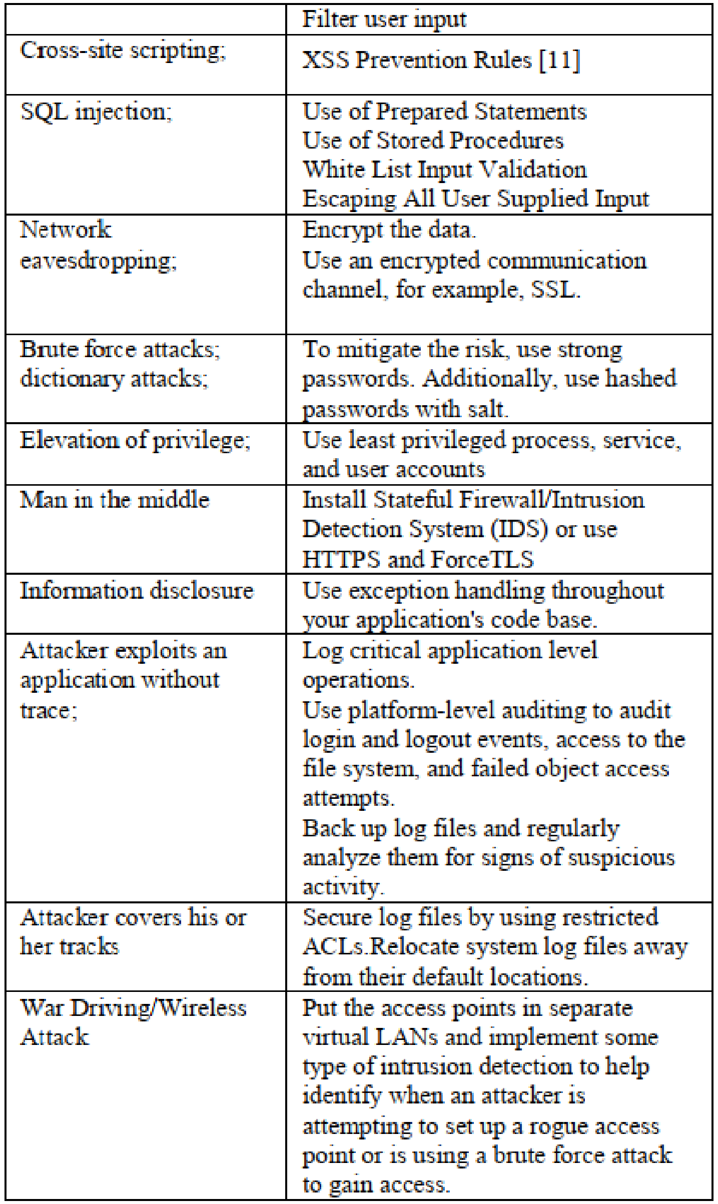


Figure 45: Connected Conservation's (2022) interoperable anti-poaching system

Appendix B

Table 18: Naagas and Palaoag's (2018) identified threats and countermeasures

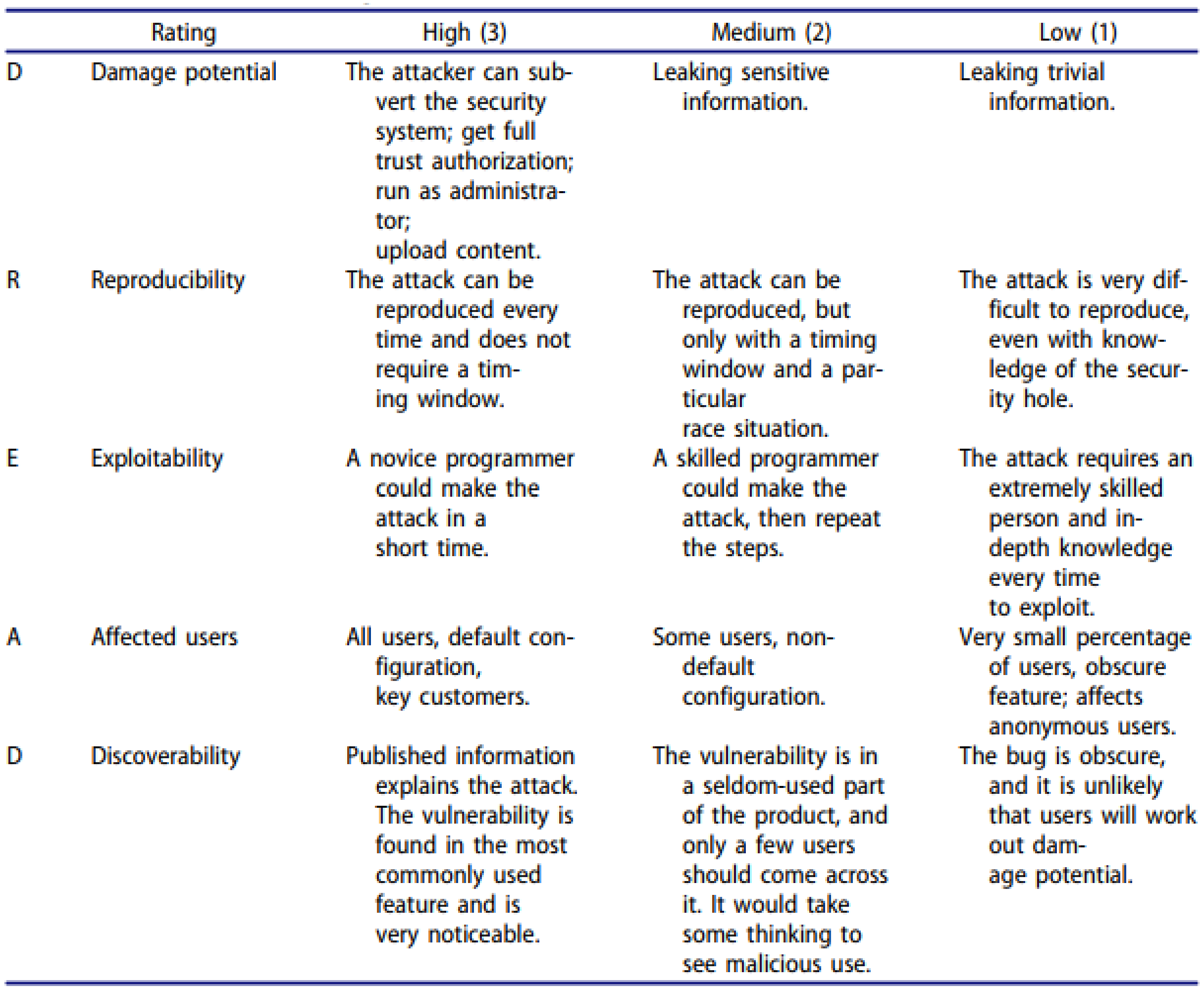




THREATS	COUNTERMEASURES
Information gathering	Configure routers to restrict their responses to foot printing requests.
Spoofing	Filter incoming packets that appear to come from an internal IP address at your perimeter. Filter outgoing packets that appear to originate from an invalid local IP address
Sniffing	Use strong physical security and proper segmenting of the network. Encrypt communication fully, including authentication credentials.
Session hijacking	Use encrypted session negotiation. Use encrypted communication channels.
Denial of service	Use a network Intrusion Detection System (IDS) because these can automatically detect and respond to SYN attacks.
Viruses	Install Anti Virus
Foot printing	Use an IDS that can be configured to pick up foot printing patterns and reject suspicious traffic.
Password cracking	Use strong passwords for all account types.
Arbitrary code execution	Configure web server to reject URLs with "../" to prevent path traversal.
Buffer overflow;	Avoid using library files
	Filter user input
Cross-site scripting;	XSS Prevention Rules [11]
SQL injection;	Use of Prepared Statements Use of Stored Procedures White List Input Validation Escaping All User Supplied Input
Network eavesdropping;	Encrypt the data. Use an encrypted communication channel, for example, SSL.
Brute force attacks; dictionary attacks;	To mitigate the risk, use strong passwords. Additionally, use hashed passwords with salt.
Elevation of privilege;	Use least privileged process, service, and user accounts
Man in the middle	Install Stateful Firewall/Intrusion Detection System (IDS) or use HTTPS and ForceTLS
Information disclosure	Use exception handling throughout your application's code base.
Attacker exploits an application without trace;	Log critical application level operations. Use platform-level auditing to audit login and logout events, access to the file system, and failed object access attempts. Back up log files and regularly analyze them for signs of suspicious activity.
Attacker covers his or her tracks	Secure log files by using restricted ACLs.Relocate system log files away from their default locations.
War Driving/Wireless Attack	Put the access points in separate virtual LANs and implement some type of intrusion detection to help identify when an attacker is attempting to set up a rogue access point or is using a brute force attack to gain access.

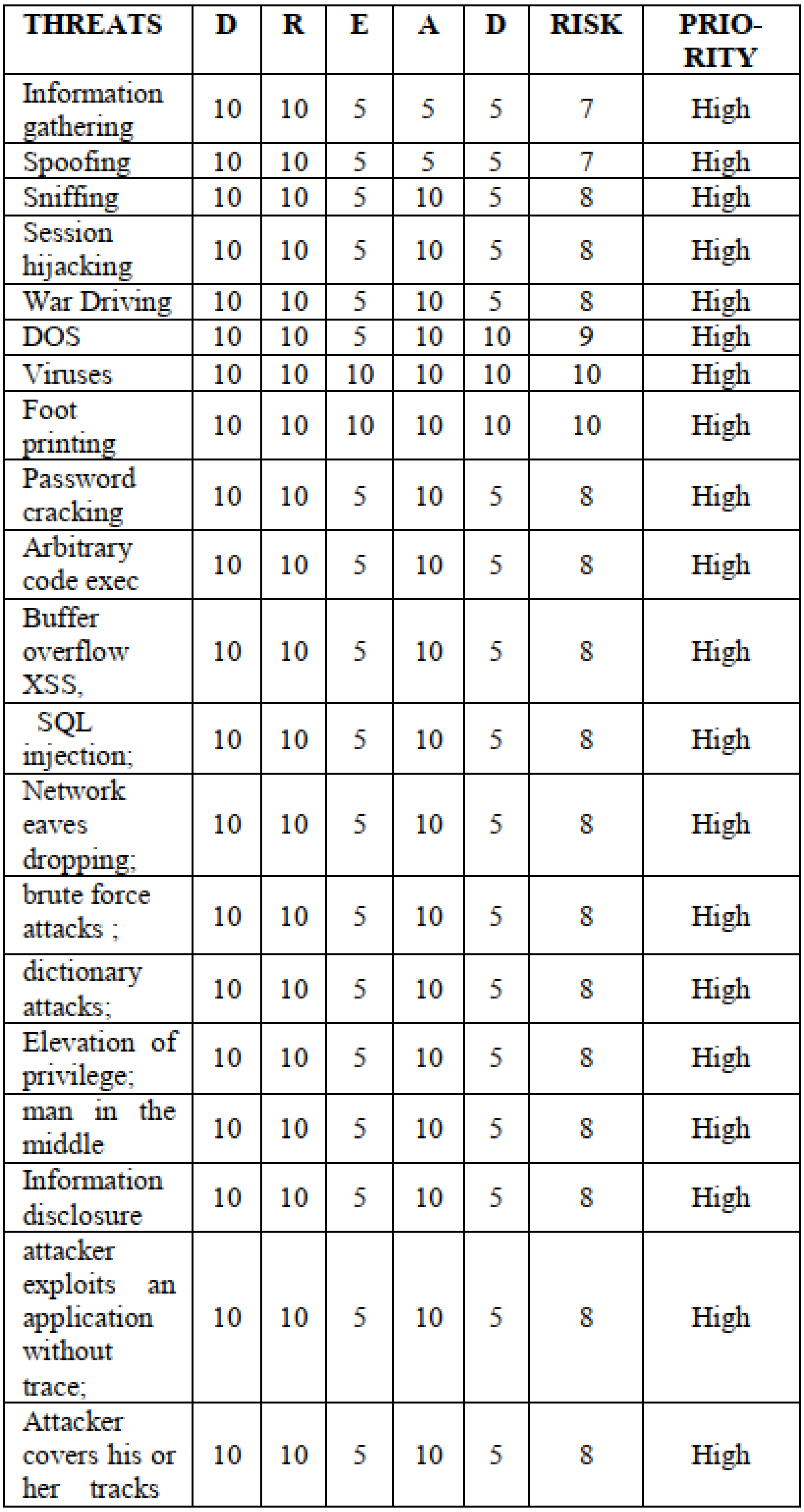
Appendix C

Table 19: Omatosho, Haruna & Olaniyi's (2019) DREAD risk levels



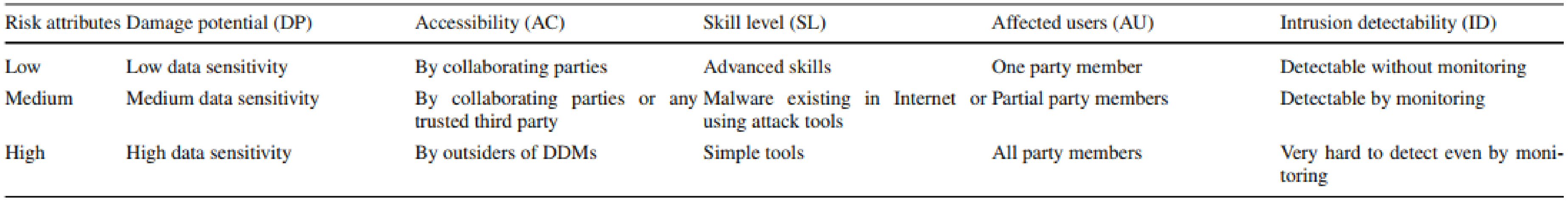
	Rating	High (3)	Medium (2)	Low (1)
D	Damage potential	The attacker can subvert the security system; get full trust authorization; run as administrator; upload content.	Leaking sensitive information.	Leaking trivial information.
R	Reproducibility	The attack can be reproduced every time and does not require a timing window.	The attack can be reproduced, but only with a timing window and a particular race situation.	The attack is very difficult to reproduce, even with knowledge of the security hole.
E	Exploitability	A novice programmer could make the attack in a short time.	A skilled programmer could make the attack, then repeat the steps.	The attack requires an extremely skilled person and in-depth knowledge every time to exploit.
A	Affected users	All users, default configuration, key customers.	Some users, non-default configuration.	Very small percentage of users, obscure feature; affects anonymous users.
D	Discoverability	Published information explains the attack. The vulnerability is found in the most commonly used feature and is very noticeable.	The vulnerability is in a seldom-used part of the product, and only a few users should come across it. It would take some thinking to see malicious use.	The bug is obscure, and it is unlikely that users will work out damage potential.

Table 20: Naagas and Palaoag's (2018) DREAD risk scoring



THREATS	D	R	E	A	D	RISK	PRIORITY
Information gathering	10	10	5	5	5	7	High
Spoofing	10	10	5	5	5	7	High
Sniffing	10	10	5	10	5	8	High
Session hijacking	10	10	5	10	5	8	High
War Driving	10	10	5	10	5	8	High
DOS	10	10	5	10	10	9	High
Viruses	10	10	10	10	10	10	High
Foot printing	10	10	10	10	10	10	High
Password cracking	10	10	5	10	5	8	High
Arbitrary code exec	10	10	5	10	5	8	High
Buffer overflow XSS,	10	10	5	10	5	8	High
SQL injection;	10	10	5	10	5	8	High
Network eaves dropping;	10	10	5	10	5	8	High
brute force attacks ;	10	10	5	10	5	8	High
dictionary attacks;	10	10	5	10	5	8	High
Elevation of privilege;	10	10	5	10	5	8	High
man in the middle	10	10	5	10	5	8	High
Information disclosure	10	10	5	10	5	8	High
attacker exploits an application without trace;	10	10	5	10	5	8	High
Attacker covers his or her tracks	10	10	5	10	5	8	High

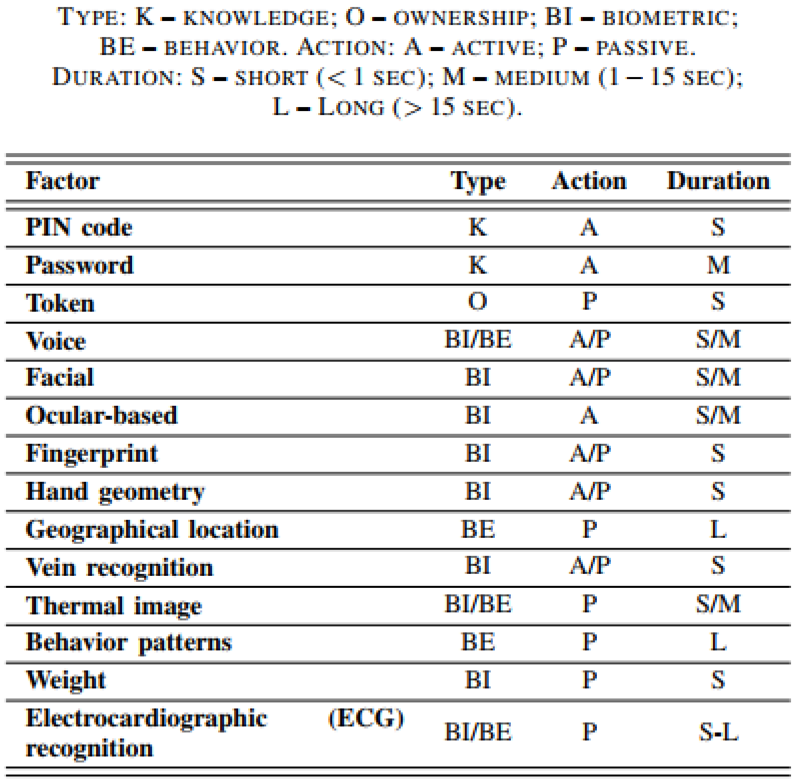
Table 21: Zhang, *et al*.'s (2021) redefined DREAD benchmarks



Risk attributes	Damage potential (DP)	Accessibility (AC)	Skill level (SL)	Affected users (AU)	Intrusion detectability (ID)
Low	Low data sensitivity	By collaborating parties	Advanced skills	One party member	Detectable without monitoring
Medium	Medium data sensitivity	By collaborating parties or any trusted third party	Malware existing in Internet or using attack tools	Partial party members	Detectable by monitoring
High	High data sensitivity	By outsiders of DDMs	Simple tools	All party members	Very hard to detect even by monitoring

Appendix D

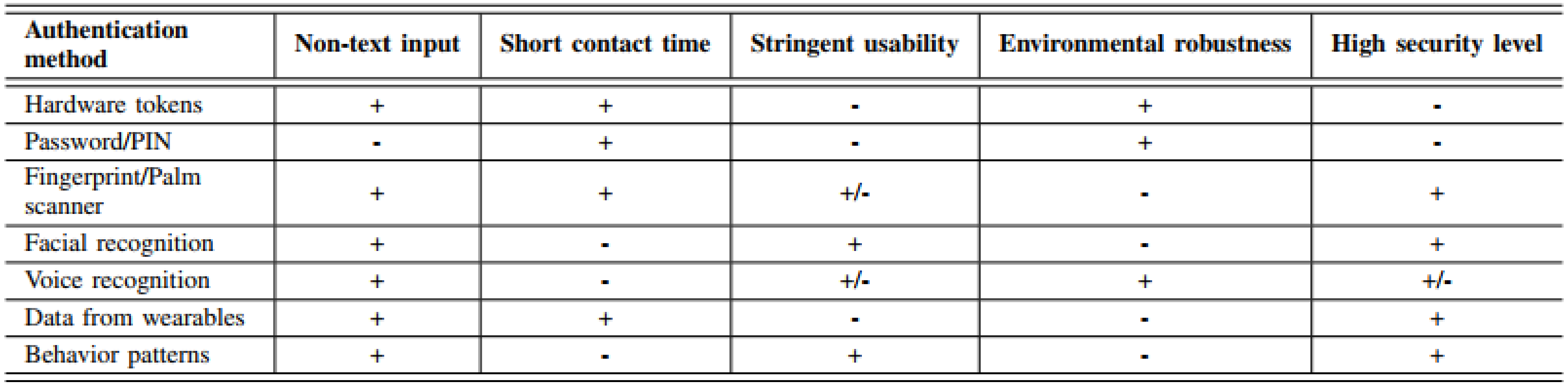
Table 22: Different authentication factors (Ometov, *et al*., 2019)



TYPE: K – KNOWLEDGE; O – OWNERSHIP; BI – BIOMETRIC; BE – BEHAVIOR. ACTION: A – ACTIVE; P – PASSIVE. DURATION: S – SHORT (< 1 SEC); M – MEDIUM (1 – 15 SEC); L – LONG (> 15 SEC).

Factor	Type	Action	Duration
PIN code	K	A	S
Password	K	A	M
Token	O	P	S
Voice	BI/BE	A/P	S/M
Facial	BI	A/P	S/M
Ocular-based	BI	A	S/M
Fingerprint	BI	A/P	S
Hand geometry	BI	A/P	S
Geographical location	BE	P	L
Vein recognition	BI	A/P	S
Thermal image	BI/BE	P	S/M
Behavior patterns	BE	P	L
Weight	BI	P	S
Electrocardiographic (ECG) recognition	BI/BE	P	S-L

Table 23: Ometov, *et al*.'s (2019) evaluation of authentication factors



Authentication method	Non-text input	Short contact time	Stringent usability	Environmental robustness	High security level
Hardware tokens	+	+	-	+	-
Password/PIN	-	+	-	+	-
Fingerprint/Palm scanner	+	+	+/-	-	+
Facial recognition	+	-	+	-	+
Voice recognition	+	-	+/-	+	+/-
Data from wearables	+	+	-	-	+
Behavior patterns	+	-	+	-	+