National University of Science and Technology
POLITEHNICA Bucharest

Faculty of Automatic Control and Computers, Computer Science and Engineering Department

PhD Thesis

in Computer Science, Information Technology and System Engineering

User-centric Cybersecurity Attacks Management in Future Generation Computer Systems

Gestionarea Atacurilor de Securitate Cibernetică centrată pe Utilizator în Sistemele Informatice de Generație Viitoare

SUMMARY

presented by

Drd.ing. Răzvan-Constantin STOLERIU

supervised by

Prof.dr.ing. Florin POP

2024
Bucharest, Romania

Contents

List of Figures

List of Tables

1 | Introduction

1.1 Research Statement and Thesis Objectives

Nowadays, as technology has evolved through advancements in cloud computing, online banking operations, social networks, and automated processes, the information that is stored, managed by various IT devices, or that passes through networks is at risk. The same accelerating rhythm in developing new tactics and techniques is observed on the adversaries' side. They launch cyber-attacks that target from regular users and companies to governments to steal data and cause damage. Cyber threats encompass attacks against critical infrastructures that include systems in transportation, communications, water, and power fields. The security threats also consist of operations conducted by cyber-criminals to defraud and steal data from victims [6]. Some of the effects of these attacks would be the disruption of business continuity and customer trust degradation [2]. The adversaries try not to leave traces on the compromised hosts by disabling or bypassing in-place security tools, removing security logs, and deleting created accounts and malware samples used during the incident.

Table 1.1 presents the effects and consequences of modern cyber threats against various domains.

Taking into account the fact that nowadays adversaries develop sophisticated malware samples to evade detections and thwart malware analysis, solid detection strategies and countermeasures procedures should be implemented. If they can not fully stop the adversaries' plan, at least they can make their mission harder to fulfill, and maybe they can leave some traces so that they can be detected afterward [4].

All detection strategies and countermeasures procedures we proposed along with the security solution we designed and developed, have determined different security approaches with different results, each presented in the following chapters. The results we have reached are based on the research questions and objectives established in Table 1.2.

Our detection solutions did not propose to build a user-centric context, but consider cases generally available for all users.

The main goal of the thesis is to identify the modern cyber attack types and develop detection strategies and security solutions that can provide accurate results and fast response time. To this respect, we started with a survey of the existing solutions in the literature and then tested and proposed new approaches that could bring innovation and better results. We compared the systems we designed with the ones that were proposed by other researchers in the field and outlined the advantages and improvements ours have.

We defined the thesis goals by answering the following research questions:

1. **What are the modern cyber-security attack types, and how could organizations defend against them? (RQ1)**

   We surveyed different research papers in the literature and blogs from popular security ven-

dors to identify the most prevalent cyber-security attacks, determine how they are operated, which are the most innovative tactics and techniques. Further, we studied the top infection vectors used by adversaries to get access to the systems and classified different malware families that targeted various industries and operating systems. We have proposed modern cyber-attack detection strategies and countermeasures that can assist regular users and researchers to defend, detect, or at least minimize the risk of a successful attack that could target them.

2. **How could malicious short URLs be detected? (RQ2)**

   Short URLs have been designed to offer the possibility of utilizing a URL that has fewer characters and points to the original site as the long URL. Recently, many security vendors reported the use of short URLs in malware campaigns. The attackers prefer this method since they can hide the real malicious website, and victims do not know what is behind the shortened form. We have proposed a combined solution that uses Machine Learning (ML) algorithms and different open-source Threat Intelligence platforms to detect malicious short URLs.

3. **How to detect malicious short URLs in smishing attacks? (RQ3)**

   According to a threat intelligence (TI) report [15], phishing was the most common infection vector used throughout 2021. Phishing has many forms, depending on how it is performed. For example, if it is achieved via SMS messages, it is called smishing (i.e., sms phishing). Smishing has been used in multiple campaigns for malware sample delivery, such as TeaBot, TangleBot, and FluBot. We have proposed a novel Cloud-Edge model to detect malicious short URLs in smishing attacks. Our designed system leverages TI platforms and Machine Learning algorithms.

4. **How to increase the security systems' efficiency in cyber attack detection? (RQ4)**

   One of the security solutions used in cyber-attack detection is Endpoint Detection and Response (EDR) systems. They are sophisticated intrusion detection systems that match system events with known adversarial behaviors. Although they promise satisfactory results in cyber-security threat mitigation, they face the following challenges: they generate a high number of false positives, the alerts' veracity must be manually verified by a security analyst, etc. Therefore, we proposed a distributed solution based on the Cloud-Edge model for smishing attack detection that diminishes the shortcomings of classical EDR systems through modern solutions such as threat intelligence and machine learning. Moreover, our designed solution integrates a ChatBot solution that facilitates ease of usage and increases users' awareness. This project we proposed was part of a *Proof of Concept* program (*Invest National University of Science and Technology POLITEHNICA Bucharest Proof of Concept* program), and we worked on the development of a business plan.

5. **How to detect cyberbullying in multimedia files? (RQ5)**

   A recently observed form of cyber attack is cyberbullying. It occurs when a person threatens, worries, or terrifies somebody through electronic means. This malicious behavior usually takes place in schools, universities, or at home. It is usually achieved via social media networks (e.g., TikTok) where people send or share videos or GIF files that have bullying content. We proposed two different solutions for cyberbullying detection. One targets GIF files, while the other handles TikTok videos.

6. **What are the main cybersecurity risks and potential countermeasures procedures for large-scale infrastructures? (RQ6)**

   Transportation, education, healthcare, government, infrastructure, and many other fields in a large-scale infrastructure passed through the technologization process. Consequently, there are a lot of programs, applications, and automated processes that are critical for business continuity and service delivery. Their importance and usability for human daily lives made attackers consider these domains as potential targets for getting more profit. We proposed a framework for cybersecurity risk assessment in large-scale infrastructures alongside mitigation techniques and countermeasures. The methodology we employed in our study is qualitative and is based on projects from the CORDIS database of the European Commission related to cybersecurity governance in smart cities.

1.2 Thesis Outline

The main objectives of this thesis are the identification of the most prevalent cyber attack types in the modern era and the development of detection strategies, countermeasures procedures, and robust security solutions that provide efficiency, accurate results, and fast response time.

The first step in accomplishing this objective is to determine the top cyber-attack types, infection vectors, malware development tendencies, and cyber threats that target various industries and operating systems. Besides this, we searched for related detection strategies and countermeasures procedures.

The specific sub-objectives are the following: malicious short URL detection, smishing attack identification, the development of a next-generation agent-based EDR system for cybersecurity threats, cyberbullying detection in multimedia files, and cybersecurity governance in large-scale infrastructures.

In Chapter 2, we present the most recent cyber-attack types, infection vectors, threat groups, malware development tendencies, and many other things of interest that describe the threat landscape for the 2021-2022 period.

In Chapter 3, we highlight traditional and modern cyber-attack detection strategies along with the latest countermeasures procedures. They provide important guidance for defending organizations against sophisticated malware samples.

In Chapter 4, we present an exhaustive system for malicious short URL detection. We integrate our system with open-source threat intelligence platforms and Machine Learning (ML) algorithms.

In Chapter 5, we discuss a scalable technique to detect malicious URLs in smishing attacks based on a Cloud-Edge architecture, using threat intelligence platforms and Machine Learning algorithms that classify the URLs based on their features. To showcase the effectiveness of our solution, we implement an Android application that detects malicious short URLs in SMS messages and notifies the user concerning their legitimacy. This chapter is based on a research article that represents a revised and expanded version of a paper entitled *Malicious Short URLs Detection Technique* presented at *2023 22nd RoEduNet Conference: Networking in Education and Research (RoEduNet), Craiova, Romania, 2023.*

In Chapter 6, we present two solutions for cyberbullying detection in multimedia files using deep learning models. The first performs bullying detection in GIFs, while the other accomplishes the same task for TikTok videos.

In Chapter 7, we discuss *NextEDR* - Next-generation agent-based EDR systems for cyber-

security threats, an innovative and interactive Cloud-Edge-Continuum Endpoint Detection and Response platform for protecting modern organizations from cybersecurity attacks. We design a Proof-of-Concept based on an interactive communication agent (ChatBot) solution for phishing detection in short URLs. This project we proposed was part of a *Proof of Concept* program (*Invest National University of Science and Technology POLITEHNICA Bucharest Proof of Concept* program), and we worked on the development of a business plan.

In Chapter 8, we present a framework for cybersecurity risk assessment in large-scale infrastructures alongside mitigation techniques and countermeasures procedures. The methodology we employ in our study is qualitative. It is based on 66 projects from the CORDIS database of the European Commission related to cybersecurity governance in smart cities.

In Chapter 9, we present the main conclusions of the research of this thesis with the main advantages and shortcomings of each of the proposed security approaches. The main focus of the thesis represents the identification of the most prevalent cyber threats in the modern era and the development of solid detection strategies, countermeasures procedures, and robust security solutions that provide high accuracy and fast response time.

Table 1.1. Cyber threats' effects and consequences against various fields.

	University	Government	Healthcare	Social Media	Bigger Chat Groups
Short URL in smishing attacks	1. If students are tricked into providing their credentials, attackers could gain access to university systems. 2. Fake pages can steal their login information.	1. If employees provide their username and password, adversaries could get access into the public institutions systems. 2. Employees can disclose confidential information from the workplace.	1. The doctors are redirected to fake medical websites. 2. Attackers could steal medical records.	1. Users receive suspect links that steal sensitive data. 2. Adversaries could send links to steal credit card information.	1. Participants are prone to scams. 2. Users can fear engaging with others.
Cyberbullying through GIF files	1. Students can lose self-esteem and start to be ashamed. 2. The ability to concentrate may be affected.	1. It can cause a decrease in work efficiency. 2. Relationships with those around are affected.	1. Patients can read harmful messages which could affect their state of health. 2. The emotional state can be affected.	1. Users receive insults on their posts. 2. It can cause social isolation.	1. Users become worried about their worth. 2. The rest of the group members could laugh at the bullied person.
Cyberbullying through TikTok videos	1. It causes the drop in academic results. 2. Students are likely to miss school.	1. It leads to the deterioration of the relationship with those around. 2. It is possible to generate misunderstandings and scandals between employees.	1. It can cause shock, and headache pain to patients. 2. Patients may refuse the doctor's consult.	1. Users' experiences become unpleasant. 2. It may cause long-term anxiety.	1. Users can feel isolated. 2. The affected persons may leave the group.
Cyberbullying through text messages	1. Increases the stress and depression levels. 2. Students may feel inferior to other colleagues.	1. Employees are no longer involved so much in work activities. 2. Employees may become more aggressive and suspicious about the ones surrounding them.	1. It can cause insomnia and fatigue in patients. 2. The patients may become more stressed, making them overeat.	1. Users receive negative comments. 2. Users may get body image concerns that lead to increased body dissatisfaction.	1. Users receive insults. 2. The other group members could amplify the victim's harassment and finally eliminate them from the group.

Table 1.2. Thesis objectives and methodology.

Research question	Research objective	Research methodology	Use case
RQ1	To identify the modern cyber-security attack types and how could organizations defend against them.	Proven by literature and security blogs review	To answer this question, we surveyed the most prevalent cyber attacks that define the threat landscape for the 2021-2022 period. We analyzed the top attack types that target various organizations and industries, the infection vectors utilized to get access to the systems, the actual malware development trends, the most active threat groups, and so on. Moreover, we proposed traditional and modern detection strategies along with the latest countermeasures procedures. Details about the obtained results could be read in chapters 2, and 3 and in paper [20].
RQ2	To detect malicious short URLs.	Proven by simulations	To address this question, we proposed an exhaustive system for malicious short URLs detection by leveraging threat intelligence (TI) data from popular platforms like VirusTotal, and PhishTank, and by employing various Machine Learning (ML) algorithms. Our system works for every URL, no matter the shortening service used either public or custom. To showcase the effectiveness of our solution, we have developed a generic detection system from scratch that identifies malicious short URLs and notifies the user concerning their legitimacy. Details about the experimental results can be read in chapter 4 and paper [19].
RQ3	To detect malicious short URLs in smishing attacks.	Proven by simulations	To provide an insightful response to this question, we proposed a scalable technique to detect malicious URLs in smishing attacks based on a Cloud-Edge architecture, using threat intelligence platforms (e.g., VirusTotal, PhishTank), and Machine Learning algorithms that classify the URLs based on their features. The model's accuracy reaches up to 97%. To showcase the effectiveness of our solution, we implement an Android application that detects malicious short URLs in SMS messages and notifies the user concerning their legitimacy. It is a revised and expanded version of the solution published in paper [19]. More details about the experimental results can be read in chapter 5.
RQ4	To increase the security systems' efficiency in cyber attack detection.	Proven by simulations	To answer this question, we proposed NextEDR, a next-generation agent-based EDR system for cybersecurity threats. It is an innovative and interactive Cloud-Edge-Continuum Endpoint Detection and Response platform for protecting modern organizations from cybersecurity attacks. We design a Proof-of-Concept based on an interactive communication agent (ChatBot) solution for phishing detection in short URLs. Our solution is a mobile-centric multi-layer platform based on the Cloud-Edge-Continuum model. This project we proposed was part of a Proof of Concept program, and we worked on the development of a business plan. The Minimum Viable Product (MVP) development for the NextEDR platform involved iterative refinement based on user feedback, prioritizing essential features to deliver a functional solution that addresses core user needs efficiently. Details about the experimental results could be read in chapter 7, and in paper [10].
RQ5	To detect cyberbullying in multimedia files.	Proven by simulations	To provide an insightful response to this question, we propose a system for cyberbullying detection in multimedia files using deep learning models. It comprises two solutions: one that performs bullying detection in GIFs and another that accomplishes the same task for TikTok videos. The first solution employs a hybrid architecture that comprises a Convolutional Neural Network (CNN) and three Recurrent Neural Networks (RNNs). The second system leverages a Transformer-based model that operates on Convolutional Neural Network (CNN) feature maps. Details about experimental results could be read in chapter 6, or in paper [18].
RQ6	To identify the main cybersecurity risks and potential countermeasures procedures for large-scale infrastructures.	Proven by literature and security blogs review	To answer this question, we proposed a framework for cybersecurity risk assessment in large-scale infrastructures alongside mitigation techniques and countermeasures. The methodology we employed in our study is qualitative and is based on projects from the CORDIS database of the European Commission that are related to cybersecurity governance in smart cities. Details about the obtained results could be read in chapter 8.

2 | Critical Analysis of Modern Cyber Security Attacks

In this chapter, we discuss the most recent cyber-attack types, infection vectors, threat groups, malware development tendencies, and many other things of interest that describe the threat landscape for the 2021-2022 period. The content of this chapter is based on the publication of the paper *Modern Cyber Security Attacks, Detection Strategies, and Countermeasures Procedures* in *2023 24th International Conference on Control Systems and Computer Science (CSCS)*.

2.1 The Actual Tendencies for Malware Development

The malware authors use their skills to build more sophisticated samples that can easily bypass the in-place security systems. In this section, we present the modern tendencies for malware development, covering the techniques that are the most employed by adversaries to achieve their final goal.

Next-level Detection Evasion

Ransomware authors changed their attack tactic by using intermittent encryption that is much faster (only some blocks of data are encrypted, not the entire filesystem). Some changes also were observed in C2 communications, adversaries making use of the cloud technology not to be uncovered. They even utilize the DNS protocol to hide their communication (a.k.a. DNS Tunneling). To hinder malware analysis, adversaries use advanced code packers (e.g., UPX, Themida, MPRESS) and obfuscation techniques. The use of different programming languages, such as PureBasic or Nim, decreases the ease of the reverse engineering process [15].

Malware Focus on Virtualization

Virtualization platforms preferred by adversaries are Docker and Windows containers or Kubernetes. Multiple ransomware families were observed targeting Linux-based VMWare ESXi servers. Instead of encrypting the operating system that runs inside, the adversaries chose to use this type of attack against the virtual machine (VM) files.

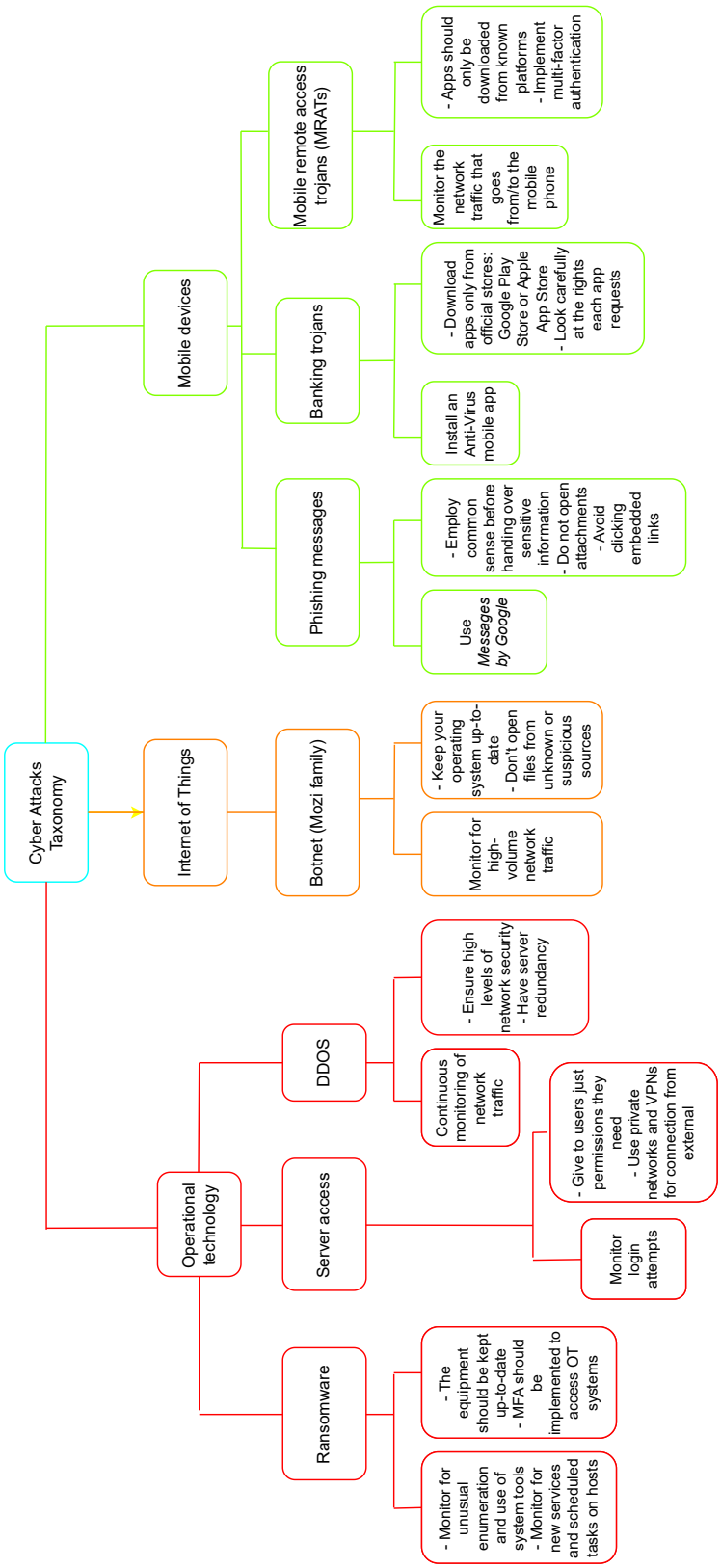

Figure 2.1. Cyber attacks taxonomy.

2.2 Current Threat Landscape

This section presents the most prevalent cyber threats in the modern era. It includes the top cyber-security attack types, infection vectors, geographic trends, the most active threat groups, and many other things of interest.

In Fig. 2.1, the taxonomy starts with a short classification of the equipment or devices targeted by adversaries. Then, the most prevalent types of attacks are exhibited for every one of these platforms. Finally, the detection strategies and countermeasures that can be employed for each kind of attack are displayed on the left and right sides, respectively.

The most prevalent cyber-security attack types observed in the modern era are ransomware, server access, business email compromise, data theft, and credential harvesting.

An attack vector represents a method by which an adversary obtains unauthorized access to a system in order to achieve malicious actions [23]. The top infection vectors used by adversaries are phishing, vulnerability exploitation, stolen credentials, brute force, and remote desktop.

According to a report published by Proofpoint, researchers have identified in Europe a rise of 500% of mobile malware delivery attempts between the start and the end of February 2022 [9]. Today's mobile malware specimens have more capabilities being able to track location, remove important files, or record phone audio and video. Table 2.1 describes the most common mobile malware specimens [9].

Table 2.1. Mobile malware types, functionality, and regional spread, 2022.

	Target OS	**App Impersonation**	**Financial Impersonation**	**Credential theft**	**Microphone and Camera**	**SMS Spreading**	**Privilege Escalation**	**Primary Geography**
FluBot	Android	Yes	Yes	Yes	No	Yes	Yes	Asia, UK, Europe
TeaBot	Android	Yes	Yes	Yes	No	Yes	Yes	UK, Europe
TangleBot	Android	No	Yes	Yes	Yes	No	Yes	North America
MoqHao	Android	Yes	Yes	Yes	No	Yes	No	Asia, Japan
BRATA	Android	Yes	Yes	Yes	No	Yes	No	UK, Europe, Latin
TianySpy	Android, iOS	Yes	Yes	Yes	No	No	No	Japan
KeepSpy	Android	Yes	Yes	Yes	No	No	No	Japan

The most affected industries by attacks in 2021 were manufacturing, oil and gas, transportation, utilities, mining, and heavy and civil engineering. Ransomware is the leader in the attack types that targeted industries with OT networks in 2021.

According to the X-Force Incident Response study [15], cybercriminals are leaders in the source of cyberattacks. They are followed by nation-state actors who had espionage and surveillance as the primary goals for conducting their actions.

The majority of the cyber security attacks that took place in 2021 were directed to Asia, with a percentage of 26% of the total number. From this region, Japan was the most attacked due to the Olympic games that took place there in the summer of 2021. The most prevalent malware family that targeted this continent was REvil ransomware. The finance and insurance industries were the most affected. Asia is closely followed by Europe and North America.

2.3 Conclusions

This chapter treats significant key aspects concerning modern cyber security threats by providing an overview of the types, most targeted platforms, and geographic regions. Also, it outlines the innovations that appeared in the field of malicious software development by attributing them to one of the most active attacking groups we present. These target different private and public organizations depending on the interests or causes for which they fight or protest.

3 | Detection Strategies, and Countermeasures Procedures for Cyber Attacks

In this chapter, we discuss traditional and modern cyber-attack detection strategies along with the latest countermeasures procedures. The content of this chapter is based on the publication of the paper *Modern Cyber Security Attacks, Detection Strategies, and Countermeasures Procedures* in *2023 24th International Conference on Control Systems and Computer Science (CSCS)*.

3.1 Traditional Cyber Attacks Detection Strategies

Traditional security systems, such as firewalls, virtual private network (VPN) applications, Anti-Virus (AV) solutions, and intrusion detection/prevention systems (IDS/IPS), are used to secure access to a specific device or network and protect sensitive data. They are based on rules or signatures to detect security threats. Below, we present the traditional security systems:

- Firewall - It is a network security device that monitors the traffic that gets into the network (inbound) and the one that gets out of the network (outbound). The firewall can allow the traffic to pass through or block it based on some security rules [8].
- VPN - It is a security solution that assures user privacy on the Internet by hiding their IP address. It also secures the Internet connections when users are navigating. When using a VPN solution, all data in transit is encrypted [22].
- Anti-Virus - It is a security solution that protects computers from various threats by blocking malicious websites and quarantining and removing dangerous files or programs. It scans workstations for malware detection and periodically receives updates with new virus signatures to identify the latest threats [3].
- IDS/IPS - Intrusion detection systems deal with traffic monitoring and analysis for threat detection. IPS solutions have the same capabilities as IDS, but they can also prevent cyber-attacks by terminating network connections or dropping packets [1].

3.2 Modern Cyber Attacks Detection Strategies

Today's cyber-attack detection systems perform either host computer analysis or network monitoring to capture data related to cyber-attacks [13]. Some popular attack detection strategies are:

- Host Intrusion Detection Systems (HIDS) - this equipment stays on a host and monitors different activities that take place there. To have better visibility into the ongoing operations, it collects log files and network data and looks at system calls, executed processes, process

trees, etc.

- Network Intrusion Detection Systems (NIDS) – this equipment can monitor the network traffic of an entire class that passes through a link and, based on that, can identify different cyber-attacks. A NIDS is installed on a physical machine that has to have enabled a port in promiscuous mode to intercept the traffic. However, it can not defend against attacks that happen on hosts, as a HIDS would do [7].

Nowadays, the most important approaches employed by systems that are used in cyber-attack detection are signature, anomaly, and specification. As a simple comparison, signature-based detection is utilized for known threats, while anomaly-based detection is employed for changes in behavior, being capable of detecting new or unknown attacks.

3.3 Modern Countermeasures against Cyber Attacks

In order to make the attacker's life harder and decrease their chances of success, some countermeasures need to be taken and implemented. Below, are presented some of the most important actual countermeasures:

- **Security Audits** – it is better to regularly conduct security audits to discover possible vulnerabilities, risks, misconfigurations, out-of-date programs or services, and so on, that can easily be exploited by an attacker. In companies, this measure ensures that customer data is protected [14]. Moreover, another feasible option would be the penetration tests where an attacking group's specific capabilities are emulated.
- **User awareness** – all of the employees in a company should be aware of the last phishing/vishing techniques employed by attackers to steal credentials, credit card information, or any other type of sensitive data. To prevent such attacks, employees should participate in different security training.
- **Up-to-date software** – one of the good practices is to keep the software up-to-date. This fact requires some in-place policies concerning the installation of software updates and security patches for the operating system, computer programs, drivers, etc. Updating software provides many benefits like bug fixing, vulnerability patching, improved performance, and the release of new features [21].
- **Endpoint Detection and Response** (EDR) – a plus value in the prevention and detection of cyber attacks can be brought by the EDR solutions that automatically monitor for, identify, remove, or quarantine malware specimens. EDR solutions collect data from endpoint and network levels and provide detection, prevention, investigation, and response capabilities.

3.4 Conclusions

Our research addresses modern cyber-attack detection strategies and countermeasures that can assist regular users and researchers to defend, detect, or at least minimize the risk of a successful attack that could target them. Our paper proposes several popular and efficient defensive procedures.

4 | Malicious Short URLs Detection Technique

In this chapter, we discuss an exhaustive system to detect malicious short URLs by leveraging threat intelligence data from popular platforms like VirusTotal and PhishTank and by employing various Machine Learning (ML) algorithms. Our system works for every URL, no matter the shortening service used either public or custom. The content of this chapter is based on the publication of the paper *Malicious Short URLs Detection Technique* in *2023 22nd RoEduNet Conference: Networking in Education and Research (RoEduNet)*. This chapter is organized as follows: first, we present in section 4.1 an introduction to the field of the studied issue. Then, section 4.2 introduces the implementation details of the proposed solution. In section 4.3, we present an analysis of the experimental results. Conclusions about the capabilities of the proposed solution in performing malicious short URL detection end our chapter.

4.1 Background

A URL shortening service is a system that takes a URL (Uniform Resource Locator) from a user and converts it into a short form. The latter represents an alias of the original one and can be shared with every person. When somebody accesses a short URL, there is firstly performed a network connection to the shortening service provider that redirects the request to the original website the user intends to access (i.e., the landing page). This process happens automatically, without user intervention. Short URLs can be used for both legitimate and malicious purposes. The main advantage for adversaries when using this kind of URL is they hide the final destination. Therefore, victims are not aware of the final landing page they are redirected to and can not spot some suspicious features that would be present in the original URL.

Our research focuses on the development of an exhaustive system for malicious short URL detection. To showcase the system's effectiveness, we will consider all types of short URLs, no matter the shortening service used, either public or custom. Moreover, we get the final URL the user will land on by tracing all HTTP Redirect connections. Further, we integrate our system with open-source threat intelligence platforms like VirusTotal and PhishTank to check the reputation of the final URL. If the scanning engines do not identify the final URL as malicious or suspicious, it is transmitted to our proposed Machine Learning model for classification. The model's accuracy reaches up to 97%.

The next section introduces the details of the implementation of the proposed solution.

4.2 Solution Design and Implementation

The system is designed for any user and utilizes a service for short URL reception. Once received, it is converted to the final form (i.e., a landing page) and sent to open-source Threat Intelligence platforms that we have integrated to provide accurate results and up-to-date information.

For the situation when the URL is not detected by the Anti-Virus engines or is not present in the phishing database, we propose an additional security layer that makes use of Machine Learning (ML) algorithms. The ML model has a very high detection accuracy since we propose an enhanced set of features combined with a balanced dataset so that data is processed in the correct conditions. The system's architecture is described in Fig. 4.1:

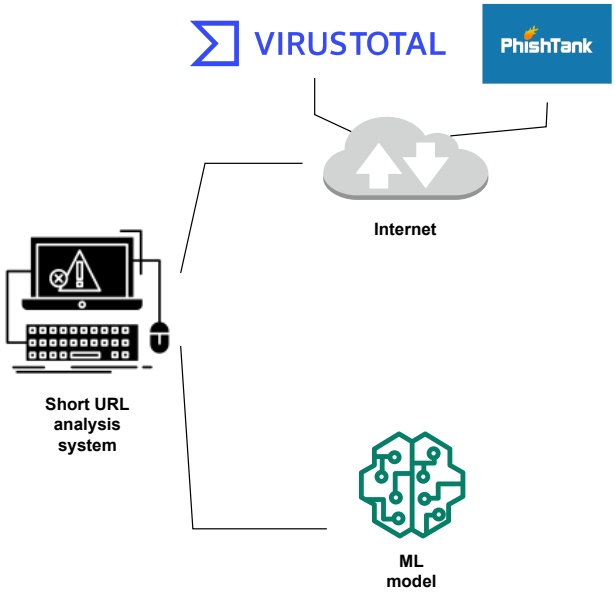

Figure 4.1. System general architecture.

Below, we present the details of each detection approach.

Threat Intelligence Approach

Fig. 4.2 presents the threat intelligence approach we have proposed in the initial security assessment of a short URL. To showcase the effectiveness of our solution, we developed a Java application using IntelliJ IDEA IDE. It listens for a short URL and forwards it to the next module. There, the final URL is obtained by connecting to the received short URL and following all HTTP Redirect connections. The final URL is sent via APIs to open-source Threat Intelligence platforms (i.e., VirusTotal and PhishTank) for scanning. If they find it malicious or suspicious, the user is notified, otherwise, the URL is sent for classification to the Machine Learning (ML) model.

Machine Learning Approach

This phase is meant to bring an additional layer of security checking to ensure the right decision is taken toward the nature of a URL. Before sending it for classification, we trained the Machine Learning (ML) model on a balanced dataset we obtained from [24]. It contains 11430 instances

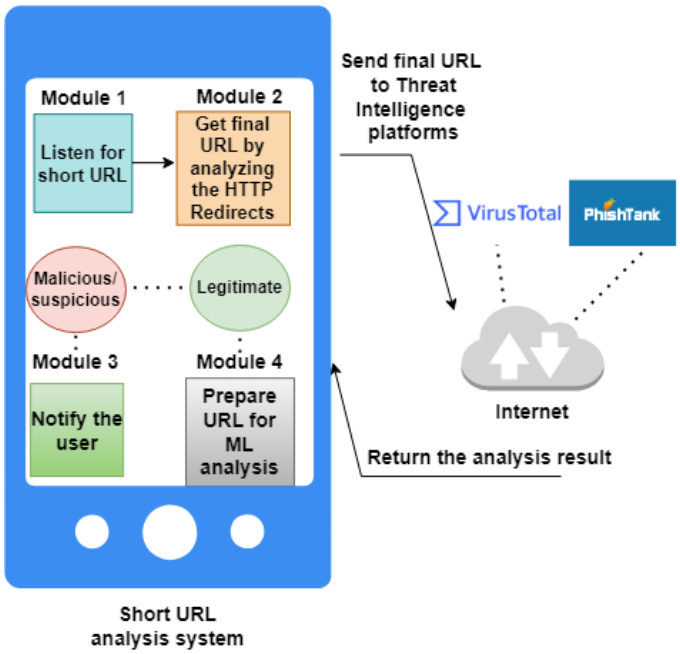

Figure 4.2. Threat Intelligence approach.

and a set of 87 features. We adapted it, added three more features, and improved its accuracy to 96.7454% using the Random Forest algorithm. These features belong to three categories: the URL lexical properties, website content, and external domain specifications.

The features that we propose belong to the URL lexical properties, and they refer to the number of parameters passed in the URL, whether it contains the '&' syntax instead of the '&' symbol, and whether a fragment is mentioned so that the user is directed in a specific zone of the webpage when they access the URL.

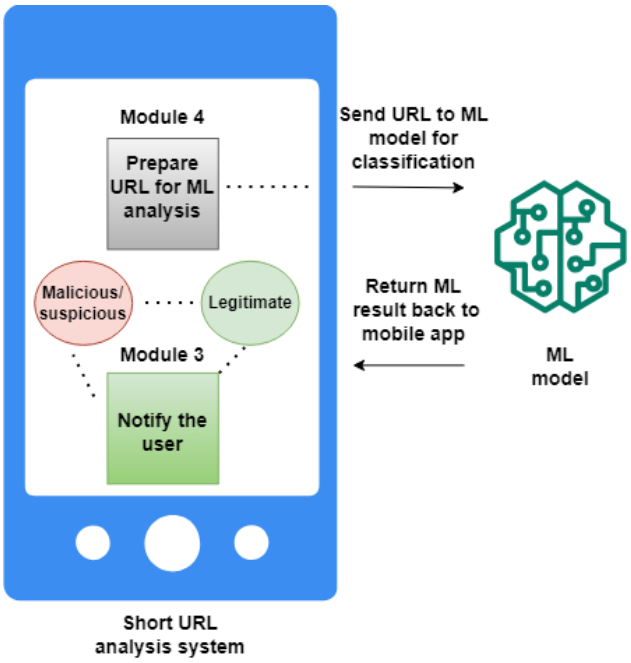

Figure 4.3. Machine Learning approach.

Fig. 4.3 describes the machine learning approach we propose as an additional layer of security checking so that an accurate decision is followed.

In the *Prepare URL for ML analysis* phase, we have the URL converted from its short form to the original one that points to the intended web destination. Further, it is sent to the ML model for classification. Once this process is done, the user is notified about the URL's reputation: malicious or legitimate.

4.3 Experimental Results and Analysis

In this section, we describe the obtained results by both the Threat Intelligence and Machine Learning (ML) approaches in malicious short URL detection. We also present details about the ML training and feature extraction processes. Finally, we compare several ML algorithms concerning their accuracy in detecting malicious short URLs.

Threat Intelligence Analysis

In this subsection, we discuss the efficiency and detection time of the Threat Intelligence (TI) approach. We have conducted some empirical tests for URLs that have been shortened and belong to the following categories: malicious, legitimate, and phishing. We took three URLs for each of these classes. More details about the short URls we conducted the tests against are presented in Table 4.1.

Table 4.1. Threat Intelligence analysis times.

Short URL	Final URL	URL Reputation	Analysis time (s)
https://shorturl.at/fuxAE	https://call.raidstore.org/	Malicious	81.267
https://t.ly/yEC_	https://technology.macosevents.com/	Malicious	2.394
https://rb.gy/w2i1u	https://press.infomapress.com/	Malicious	1.889
https://rb.gy/wtkns	https://stiri.botosani.ro/	Legitimate	1.737
https://tinyurl.com/kyc44ft8	https://www.bbc.com/	Legitimate	2.717
https://cutt.ly/zwr2HIaS	https://www.amazon.com/b?node=21576558011&ref_=alxcom_lrnmore_btn_23	Legitimate	2.864
https://urlis.net/b94wy861	https://sucursalcentroapp.brizy.site/	Phishing	2.681
http://tiny.cc/tzb8vz	https://kuccoiieeiinelovuiie.godaddysites.com/	Phishing	2.499
https://url1.io/s/VpEwo	https://anime-info777.com	Phishing	3.879

Machine Learning Model Training and Classification

In this subsection, we discuss different time metrics concerning Machine Learning (ML) training, feature extraction, and classification phases. To test the speed of the combined feature extraction and ML classification processes, we chose ten URLs for each of the categories: phishes, legitimate, and unknown. The time for these two combined processes can vary from 2.8171 (s) to 118.9281 (s), as can be seen in Table 4.2 since the feature extraction process depends on factors like the size of the website's content, the response time from other queried platforms (e.g. Open PageRank for the domain popularity, iPTY for the domain age).

Table 4.2. Duration metrics for the ML model training, feature extraction, and ML classification.

Opeartion	Duration (s)
ML model training	8.3454
Feature extraction + ML classification	Min: 2.8171
	Max: 118.9281
	Avg: 12.2929
ML classification	Min: 2.4097
	Max: 3.3585
	Avg: 2.6234

The Accuracy of the Machine Learning Classification Algorithms

We surveyed multiple papers in the literature that address the studied issue, and we observed the most prevalent machine learning (ML) classifiers employed by other researchers are JRip, PART, J48, and Random Forest. JRip and PART are rule-based classifiers, while J48 and Random Forest are based on decision trees. We did not choose complex ML algorithms such as convolutional neural networks since we do not have so much data to be classified. Instead, they are usually utilized for image or video classification. Another reason for not choosing complex ML algorithms that require much computational resources is that we would like to deploy our detection solution on any computer.

We ran four Machine Learning (ML) classification algorithms on our dataset: JRip, PART, J48, and Random Forest. The best accuracy was achieved with the latter. Table 4.3 shows the accuracy of each of the algorithms before and after our proposed features were integrated into the ML model.

Table 4.3. The accuracy of the ML algorithms.

ML algorithm	Accuracy with proposed feature set (%)	Accuracy with default feature set (%)
JRip	94.6194	94.5757
PART	95.4243	94.8206
J48	94.7332	94.5932
Random Forest	96.7454	96.6054

4.4 Conclusions

This chapter presents a combined solution that uses Machine Learning (ML) algorithms and different open-source Threat Intelligence platforms to detect malicious short URLs. The system is designed for any user and covers different scenarios: legitimate, suspicious, and malicious URLs that are shortened. The results showed that malicious short URLs can be identified, and the user is notified concerning the URL's legitimacy.

Our system analyzes any short URL, no matter the shortening services used, either public or custom.

The accuracy of the ML model is almost 97% using the Random Forest algorithm.

5 | Distributed Malicious URL Detection Technique for Smishing Attacks

In this chapter, we discuss a distributed technique to detect malicious URLs in smishing attacks based on a Cloud-Edge architecture, using threat intelligence platforms (e.g., VirusTotal, PhishTank) and Machine Learning algorithms that classify the URLs based on their features. To showcase the effectiveness of our solution, we implement an Android application that detects malicious short URLs in SMS messages and notifies the user concerning their legitimacy. The content of this chapter is based on the publication of the paper *Scalable Malicious URL Detection Technique for Smishing Attacks* in *International Journal of Computational Science and Engineering*. This chapter is structured as follows: Section 5.1 presents an introduction to the field of the studied issue. In Section 5.2, we present our proposed solution. Section 5.3 presents the deployment of our solution showcasing its effectiveness, and in Section 5.4, we highlight the experimental setup and analyze the obtained experimental results. Finally, in Section 5.5, we conclude the results of the solution and identify future research opportunities.

5.1 Background

The most frequent method to get access to smartphones and target as many people as possible is SMS phishing (e.g., smishing) [17], [16]. According to the 2022 Cyber Attack Trends: Mid-Year report from Check Point [12], FluBot is the second in the top mobile malware globally. It is an Android banking malware that uses smishing as an infection vector. Furthermore, it utilizes the same SMS message that is sent to all contacts from the victim's agenda, leading to an exponential spread.

In this paper, we present a distributed solution based on the Cloud-Edge model for smishing attack detection that diminishes the shortcomings of classical EDR systems through modern solutions such as threat intelligence and machine learning. This result is based on our previous research related to malicious short URL detection [19]. That study proposes an exhaustive system to detect malicious short URLs by leveraging threat intelligence data from popular platforms like VirusTotal and PhishTank and by employing various Machine Learning (ML) algorithms.

5.2 Proposed Architecture

In this section, we describe the proposed architecture along with its components. We present its advantages and the gains in terms of security, scalability, and efficiency end users would obtain when utilizing it.

We design a distributed architecture based on the **Cloud-Edge** model that aims to reduce the communication overhead through communication gateways (e.g., Set-top boxes from the Cloud-

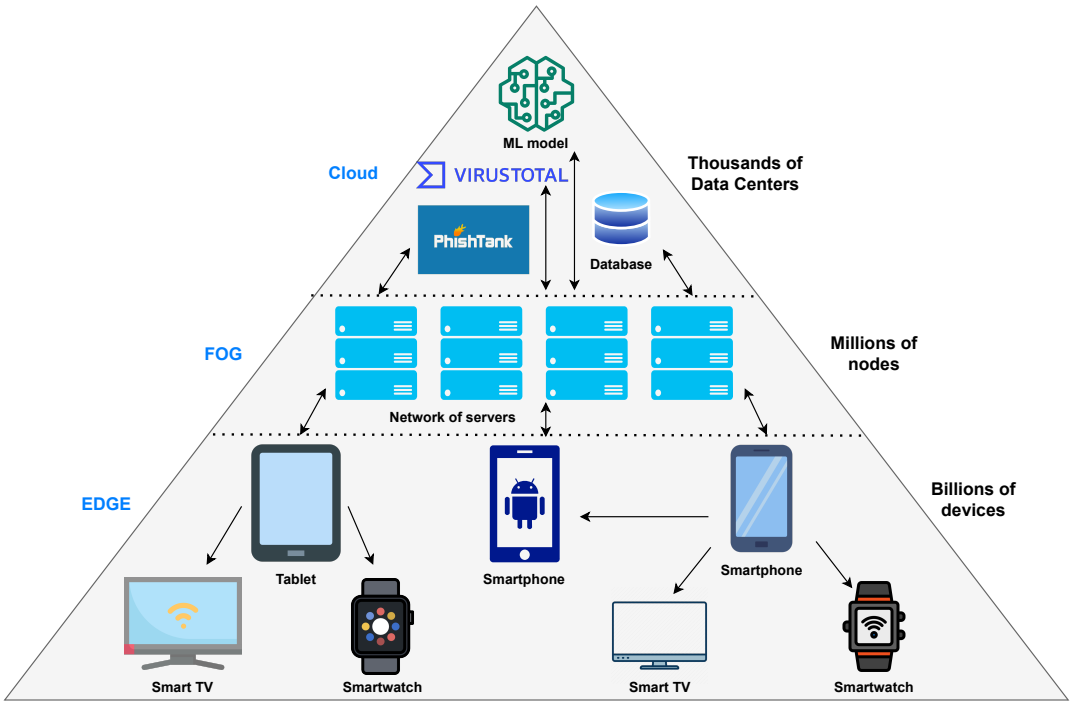

Figure 5.1. Cloud-Edge architecture.

Edge model) as presented in Figure 5.1. In the **Edge** layer, we have mobile devices such as smartphones, smartwatches, tablets, and smart TVs. Furthermore, in the **Fog** layer, we propose a solution based on Set-top boxes that are connected to the Edge and Cloud layers. The main objective of the Fog devices is to receive URL verification requests from the Edge devices and send them to the **Cloud** layer, where they are analyzed using the two following approaches:

- the Intelligence approach that is based on threat Intelligence platforms (e.g., VirusTotal and PhishTank) and,
- the Machine Learning approach that uses classification algorithms (e.g., JRip, PART, J48, and Random Forest).

In case of a positive result from the Cloud layer, the Set-top boxes underlay will send notifications to all the mobile devices connected to them. Using this approach, we ensure a low communication overhead of the system and low incident response time in case of a smishing attack.

Furthermore, our architecture is also suitable for high availability (e.g., 24/7) of incident response systems. We use load-balancing technology such that if a Set-top box fails, it is automatically replaced by another. Another important aspect we handle is the overhead problem. When there are too many requests to a Set-top box, some will automatically be handled by another. That would bring performance and efficiency.

The communication between layers is realized through Internet protocols, while the Edge devices can communicate with each other using Internet protocols or ad-hoc communication channels based on Bluetooth (e.g., Bluetooth low energy). In this way, the system guarantees the delivery of the notifications that indicate infected URLs. Therefore, in the case of a security incident, one Edge device can rapidly announce the others so that they can enable the protection measures.

The workflow is presented in Figure 5.2.

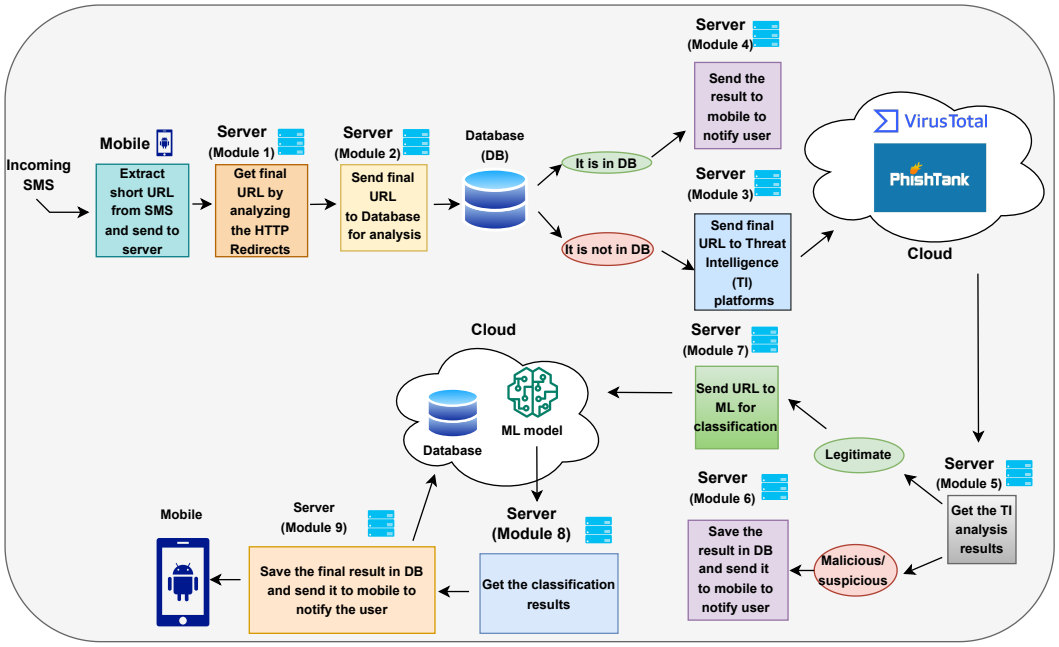

Figure 5.2. Overall workflow diagram.

5.3 System Deployment

In this section, we present details about the system deployment phase of our proposed solution. To showcase the effectiveness of our system, we ran several tests with different types of SMS messages.

- An SMS containing a malicious URL, according to Threat Intelligence data - The user receives an SMS message with a short URL. After the final URL is obtained, the Threat Intelligence platforms report it as being malicious. Fig. 5.3 and 5.4 describe these steps.

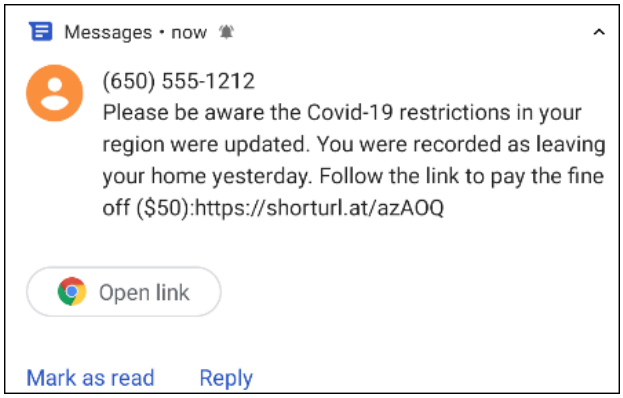

Figure 5.3. SMS with malicious short URL received.

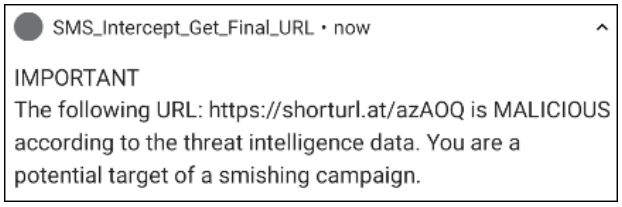

Figure 5.4. SMS successfully classified as smishing.

- An SMS containing a legitimate URL according to our ML model - The user receives an SMS message with a short URL. After the final URL is obtained, the Machine Learning (ML) model classifies it as legitimate. Fig. 5.5 and 5.6 present these phases.

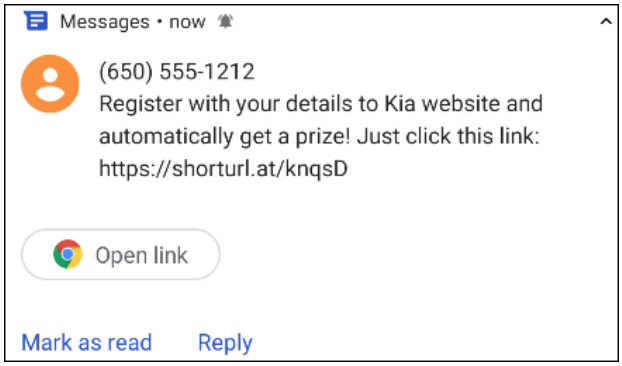

Figure 5.5. SMS with legitimate short URL received.

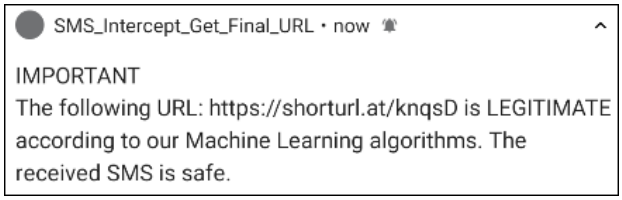

Figure 5.6. SMS successfully classified as safe.

- An SMS containing a malicious URL according to our ML model - The user receives an SMS message with a short URL. After the final URL is obtained, the Machine Learning (ML) model classifies it as malicious. Fig. 5.7 and 5.8 present these steps.

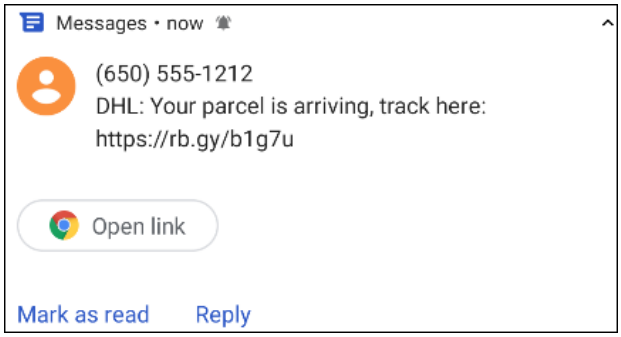

Figure 5.7. SMS with malicious URL received.

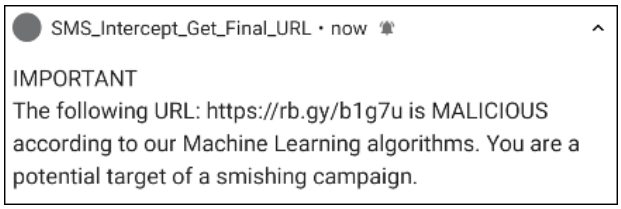

Figure 5.8. SMS successfully classified as smishing.

5.4 Experimental Results and Analysis

In this section, we describe different time and performance metrics of both the Threat Intelligence and Machine Learning approaches.

Threat Intelligence Analysis

We have conducted some empirical tests and obtained the results in Table 5.1 for the case when the URLs were not in the local database.

Table 5.1. The analysis time of the Threat Intelligence approach.

Short URL	Final URL	T.I. analysis time (s)	URL reputation
https://shorturl.at/fuxAE	https://call.raidstore.org/	81.267	Malicious + Time out from its IP
https://t.ly/yEC	https://technology.macosevents.com/	2.394	Malicious + NXDOMAIN
https://rb.gy/w2i1u	https://press.infomapress.com/	1.889	Malicious + NXDOMAIN
https://rb.gy/wtkn	https://stiri.botosani.ro/	1.737	Legitimate
https://tinyurl.com/kyc44ft8	https://www.bbc.com/	2.717	Legitimate
https://cutt.ly/zwr2HIaS	https://www.amazon.com/b?node=21576558011 &ref_=alxcom_lrnmore_btn_23	2.864	Legitimate
https://urlis.net/b94wy861	https://sucursalcentroapp.brizy.site/	2.681	Phishing + Online
http://tiny.cc/tzb8vz	https://kuccoiieeiinelovuiie.godaddysites.com/	2.499	Phishing + Online
https://url1.io/s/VpEwo	https://anime-info777.com/	3.879	Phishing + Online

Machine Learning Model Training and Classification

For the speed test of the combined feature extraction and ML classification processes, we chose ten URLs for each of the categories: phishes, legitimate, and unknown. Table 5.2 presents the obtained results. It can be observed the time for the combined processes can differ, from 2.8171 (s) to 118.9281 (s), since the feature extraction process depends on factors like the size of the website's content, the response time from other queried platforms (e.g., Open PageRank for the domain popularity, iPTY for the domain age).

Table 5.2. The duration of the feature extraction and ML classification for various types of URLs.

Operation	Item	Duration for feature extraction + ML classification (s)	Duration for ML classification (s)
Testing VALID phishes	https://vxw2.mengzhan45.top /go/?to= https://pub-c479da8c0e2748d0a34 fd7266d91fc30.r2.dev/index.html	7.8176	2.5128
	https://actividadbancaria– giovannyquint.repl.co/	6.7961	2.4780
	https://highlight-himself.toshibanetcam.com/	3.2873	2.4558
	https://dev-asistenonlibcr.pantheonsite.io/	2.8887	2.5539
	https://homeless-hospital.otzo.com/	4.6097	2.5590
	https://joint-knowledge.instanthq.com/	4.3833	2.7674
	https://department-depict.mrface.com/	4.2344	2.6300
	https://japanese-joint.qpoe.com/	4.9541	2.5126
	https://approximately-arab.mrbasic.com/	3.5512	2.5296
	https://administration-adopt.toythieves.com/	3.5949	2.4420
Testing UNKNOWN phishes	https://ameli-france-connect.com/	3.6744	2.6018
	https://portalemydati.online/	19.7659	2.4354
	http://portalemydati.online/	18.7333	2.5411
	http://infomydati.online/	19.9179	2.6577
	https://pagamenti.staffasestenza.co/8328-1/	2.8171	2.6222
	https://bafkreigfxcytcx7pfkvio4svcgeuwdpv dyey3fybfpckka7cxmhjcr6qbe.ipfs.dweb.link	8.1313	2.6082
	http://updatetan-sp.de/anmelden	4.1362	2.7528
	https://www.gefhuloa.com/pl/ pl_dfertz/?uclick=9lpmgmvc& uclickhash=9lpmgmvc-9lpmgmvc- 52ft-0-c81z-5m7vfe-5mbzi4-9e712d	3.2091	2.6611
	https://mail.kinepolis.com/optiext /optiextension.dll?ID=6yz6yKcj%2BELF jsP9CS7eIM_Z1YzAB9%2B5w3xcTWiaQ7 cRM%2BVkYi5dQ9vpKT9vUN03dFBC 0qCK46AQgv0vb_obTAikgCssL	3.5446	2.7708
	https://khatampanjereh.com/ wp-includes/knab.php	7.4725	2.6577
Testing NOT phishes	https://www.cloudflare.com/	8.9051	3.3585
	https://fonts.googleapis.com/css? family=Noticia+Text:400,400i,700,700i	4.5525	2.6238
	https://express.adobe.com	4.5663	3.0263
	http://www.paypal-merchant.com/	22.8981	2.5208
	https://www.mercadopago.com.br/	42.4390	2.4097
	https://onedrive.live.com/about/es-us-signin/	5.7948	2.5991
	http://url.zp.edu.ua/	4.8289	2.5182
	https://bitflyer.com/en-us/	11.8539	2.4980
	https://phishtank.org/phish_detail.php? phish_id=8153631	6.5013	2.9479
	https://quttera.com/	118.9281	2.44898

The Accuracy of the Machine Learning Classification Algorithms

We ran the following Machine Learning (ML) classification algorithms on our dataset: JRip, PART, J48, and Random Forest. While Random Forest outperforms the results obtained with other classifiers, we chose to highlight in Table 5.3 additional evaluation metrics.

Table 5.3. Evaluation metrics for the Random Forest classifier.

	TP Rate	**FP Rate**	**Precision**	**Recall**	**F-Measure**	**MCC**	**ROC Area**	**PRC Area**	**Class**
	0.969	0.034	0.966	0.969	0.967	0.935	0.994	0.994	legitimate
	0.966	0.031	0.969	0.966	0.967	0.935	0.994	0.994	phishing
Weighted Avg.	0.967	0.033	0.967	0.967	0.967	0.935	0.994	0.994	

5.5 Conclusions

In this chapter, we describe a novel Cloud-Edge model to detect malicious short URLs in smishing attacks. We design a multi-layer platform that reduces communication overhead and guarantees low latency and fast response time, as well as high availability through load-balancing technology. The proposed design uses Machine Learning (ML) algorithms and open-source Threat Intelligence platforms (TI) to detect malicious short URLs in smishing attacks.

Our system is designed for smartphone devices and covers different scenarios: legitimate, suspicious, and malicious URLs that are shortened. The results showed that malicious short URLs can be identified, and the corresponding SMS is classified as smishing. Our approach has the advantage of using free and open-source tools that offer an efficient system for malicious short URL detection.

Moreover, for next-level detection, our proposed Machine Learning model and enhanced set of features successfully identify the malicious URLs with an accuracy of almost 97% using the Random Forest algorithm.

6 | Cyberbullying Detection Solution for Multimedia Files using Deep Learning based Models

In this chapter, we discuss two solutions for cyberbullying detection in multimedia files using deep learning models. The first performs bullying detection in GIFs, and another accomplishes the same task for TikTok videos. The first system employs a hybrid architecture that comprises a Convolutional Neural Network (CNN) and three Recurrent Neural Networks (RNNs). The obtained results give an accuracy of 99%. The second system leverages a Transformer-based model that operates on Convolutional Neural Network (CNN) feature maps. We evaluated the model accuracy and observed we got an accuracy of up to 100%.

The goal here was not to define the context relative to the user because what cyberbullying means to me may seem like natural, normal behavior to someone else. The proposed solutions identify the action represented in a multimedia file, and we associate cyberbullying with violence. We classify a multimedia file as bullying when the action that is represented denotes a form of violence that takes place in different conditions and which gets out of the usual or normal contexts (e.g., shooting).

The cyberbullying part was one of the major directives of the European Union in the Horizon program in 2023 when it was proposed, and Romania adopted in the field of research on intelligent specialization, the prevention of cyberbullying through various methods. The research strategy for the intelligent development of Romania presents, in domain 6, point 6.4, the detection and prevention of cyberbullying and the creation of a safer Internet.

The content of this chapter is based on the publication of the papers *Bullying Detection Solution for GIFs Using a Deep Learning Approach* and *Cyberbullying Detection on TikTok Using a Deep Learning Approach* in *Information 2024* and *Sci. Bull. Univ. Politeh. Buchar.*

This chapter is structured as follows: in Section 6.1, we describe the sources used to create the datasets for both systems and present the data used in the training and testing processes. Section 6.2 highlights the architecture of the proposed systems. Section 6.3 presents the classification results of proposed solutions. Finally, in Section 6.4, we conclude the results of the proposed solutions and identify future research opportunities.

6.1 Data Collection and Labelling

In this section, we present the sources used to collect data and how it has been used for the training and testing processes. We first highlight these procedures for the system utilized in bullying detection for GIF files and then for the one leveraged in cyberbullying detection in TikTok videos.

Data Collection and Labelling for Bullying Detection Solution on GIFs

To create our dataset, we first used the UCF101 one (www.crcv.ucf.edu). We have taken videos just from the following categories: *Handstand Pushups*, *Pull Ups*, *Rowing*, and *Kayaking*. We have combined the *Handstand Pushups* and *Pull Ups* categories under *Bodybuilding*, while the *Rowing* and *Kayaking* ones under *Water sports*. These videos represent normal human activities and can be categorized as non-bullying materials.

As regards the bullying media files, we have used the GIPHY Scraper from Scrapera (https://github.com/DarshanDeshpande/Scrapera) to obtain GIFs that are associated with bullying activities. In this research, we utilize dynamic GIF files. They are images represented in the form of an animation.

Our dataset comprises 512 media files we collected from the previously mentioned sources: UCF101 dataset and Giphy. We have used 80% of data for training and 20% for testing.

The labeling for the non-bullying media files was automatic since they were taken from the UCF101 dataset that contains already labeled videos for various human actions. The labeling for the bullying media files was manually done by the authors of this paper and an outside expert.

Data Collection and Labelling for Cyberbullying Detection on TikTok videos

To create our dataset, we collected videos from TikTok that belong to the following categories: *Basketball*, *Football*, *Playing cello*, *Playing guitar*, *Shoot* and *Kick*. We have labeled the last two categories as bullying since they have content that may threaten, terrify, or worry somebody who receives and watches such a video. The remaining categories were labeled as non-bullying. We have manually downloaded all of the videos using the SnapTik platform (https://snaptik.app/). All of the videos in our dataset have between 5.5 and 6.5 seconds. If they were initially larger, we split them into media files of that range of lengths by using the Clipchamp website (https://app.clipchamp.com/).

80% of the data was used for training, while the rest was utilized for testing purposes.

6.2 Overview of Proposed Approach

In this section, we present the architecture of the proposed systems for bullying detection in GIF and TikTok video files.

Overview of Proposed Approach for Bullying Detection Solution on GIFs

We propose a bullying detection solution for GIFs. We employ a hybrid architecture that comprises a CNN and three RNNs. This architecture learns the GIFs' representation to classify them into one of the following categories: bullying and non-bullying.

The code of our proposed solution for bullying detection in GIFs using a deep learning approach is based on the following implementation from GitHub (https://github.com/keras-team/keras-io/blob/master/examples/vision/video_classification.py). Table 6.1 compares our proposed architecture with the one in the literature we took from GitHub and improved.

Table 6.1. Comparison between our proposed solution and the initial one

	Architecture	**Image size**	**Epochs**	**Training and testing datasets automatically generated**
Initial GitHub solution	1 CNN + 1 RNN	224	10	No
Proposed and improved solution *	1 CNN + 3 RNNs	169	50	Yes

Below, we discuss the architecture's main components and present how it works.

System General Architecture

Figure 6.1 presents the general architecture of the system.

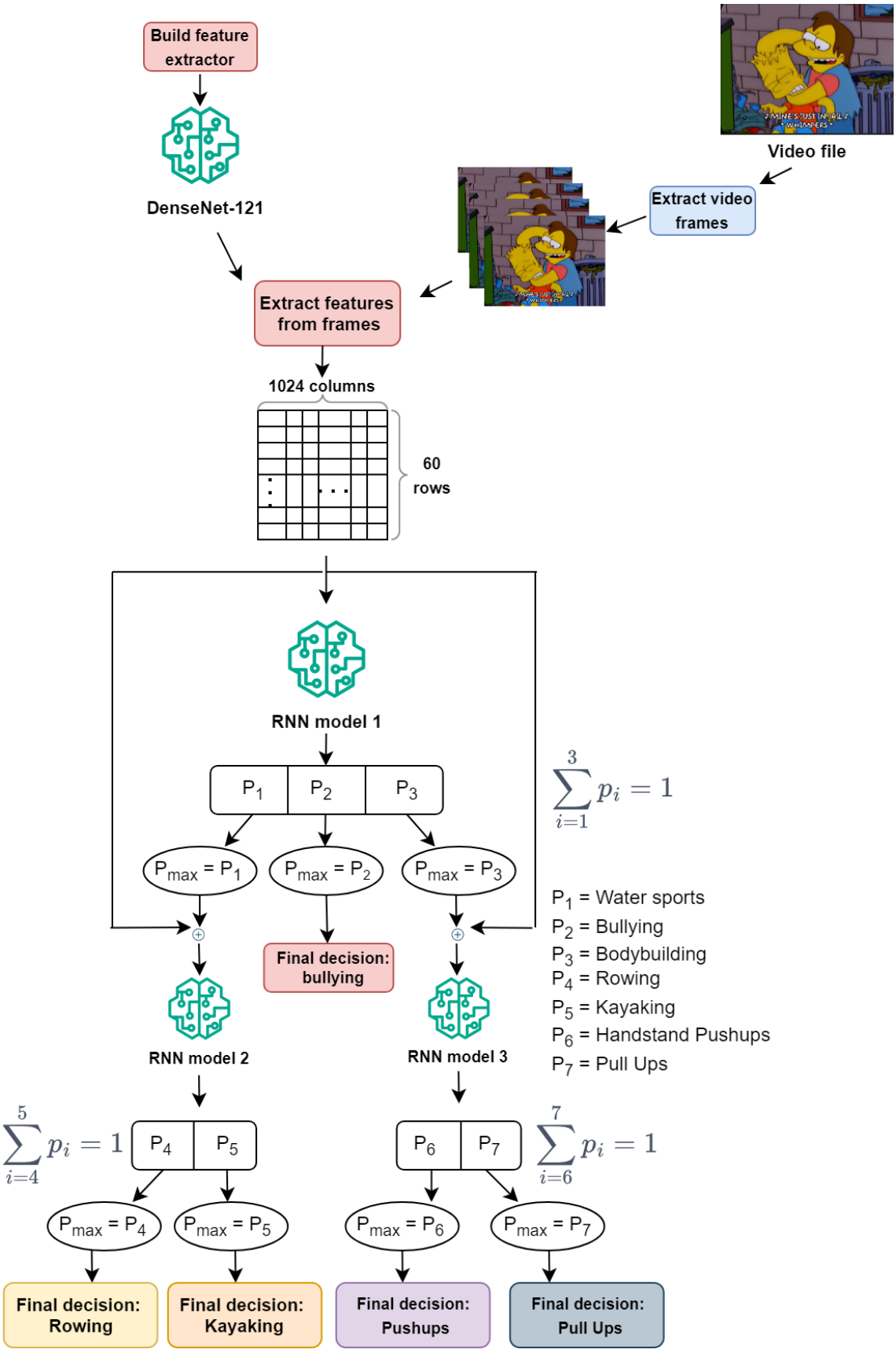

Figure 6.1. Overview of the proposed approach.

First, a feature extractor is built to get the characteristics of each video frame. It is based on a DenseNet-121 network. Second, each video file in the dataset is decomposed into frames that pass through the feature extractor so that their characteristics are obtained. These are stored in a matrix of 1024 columns and 20 rows. The first dimension, 1024, represents the number of features it uses, while the other size, 20, represents the number of frames it takes from each video. This matrix represents the input to the first RNN model trained on the entire dataset that comprises videos from the *bullying*, *Water sports*, and *Bodybuilding* categories. If the highest probability is P_2, then the evaluated video file is classified as *bullying*. Otherwise, whether the highest probability is P_1, the feature matrix is further used as input for the second RNN model trained on the *Water sports* group. In this case, the model output will be an array with two probabilities corresponding to the *Rowing* and *Kayaking* categories. On the other hand, whether the highest probability is P_3, the feature matrix is further used as input for the third RNN model trained on the *Bodybuilding* group. For this situation, the model output will be an array of two probabilities corresponding to the *Handstand Pushups* and *Pull Ups*.

Overview of Proposed Approach for Cyberbullying Detection on TikTok videos

We propose a cyberbullying detection solution for TikTok videos using a deep-learning approach. We employ a Transformer-based model that operates on Convolutional Neural Network (CNN) feature maps. Moreover, we utilize the DenseNet121 model pre-trained on the ImageNet-1k dataset.

The program we created for video classification using a deep-learning approach is based on the following code from GitHub (https://github.com/keras-team/keras-io/blob/master/examples/vision/video_transformers.py).

Table 6.2 compares our proposed architecture with the one in the literature we took from GitHub and improved.

Table 6.2. Comparison between our proposed solution and the initial one

	Feature extraction model	**Classification model**	**Maximum number of frames**	**Image size**	**Epochs**	**Training and testing datasets automatically generated**
Initial GitHub solution	DenseNet-121	Transformers-based	20	128	5	No
Proposed and improved solution *	DenseNet-121	Transformers-based	30	500	20	Yes

The system's general architecture is presented in Figure 6.2.

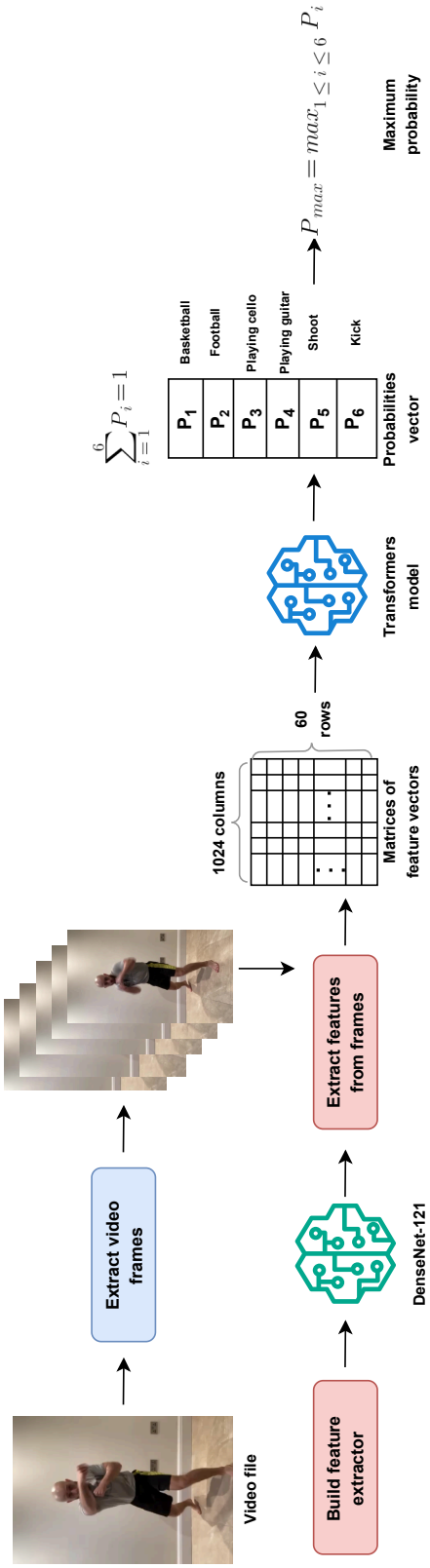

Figure 6.2. System's general architecture.

After loading the frames of the videos, we employ a convolutional neural network (CNN) to extract the features. To achieve this task, we have chosen a DenseNet architecture. In our case, the input for the DenseNet model, namely DenseNet121, are images of size 500x500, and the output for each image is a 1024 feature vector. The latter will be the input for the transformer model. To obtain the output of the CNN, we didn't include the classifier, and we only used the feature extractor. The number associated with the model denotes how many layers the model has, in this case, 121 layers.

After obtaining the feature data from DenseNet121, we employed a transformer architecture to create our bullying detector model.

6.3 Classification Results

In this section, we present the classification results of the proposed bullying detection systems. One of the platforms targets GIF files, while another deals with TikTok videos.

Classification Results for Bullying Detection Solution on GIFs

We compared the proposed solution, which has one CNN and three RNN models, with the one that has one CNN and one RNN model. Table 6.3 highlights how the solution we designed is much more efficient than the other we compared with in terms of different performance metrics. In the table, we encoded the name of each category to a symbol, as follows: *Bullying* to *(B)*, *Bodybuilding* to *(BB)*, *Water sports* to *(WS)*, *Kayaking* to *(K)*, *Rowing* to *(R)*, *Pull Ups* to *(Pull)*, and *Handstand Pushups* to *(Push)*. Moreover, in the table first, there are displayed performance metrics for the three proposed RNN models (i.e., Proposed RNN model no. 1, Proposed RNN model no. 2, Proposed RNN model no. 3) and, finally, there are described the results for the model that uses one single RNN model (i.e., Simple RNN). The architecture that employs one CNN and one RNN model has the following five classes: *Bullying*, *Kayaking*, *Rowing*, *Pull Ups*, and *Handstand Pushups*.

Table 6.3. Performance metrics for the RNN models.

RNN model no.	Accuracy	Precision	Recall	F1-Score
*Proposed RNN model no. 1	99.02%	95.24% (B) 100% (BB) 100% (WS)	100% (B) 97.37% (BB) 100% (WS)	97.56% (B) 98.66% (BB) 100% (WS)
*Proposed RNN model no. 2	97.7%	95.65% (K) 100% (R)	100% (K) 95.45% (R)	97.77% (K) 97.67% (R)
*Proposed RNN model no. 3	100%	100% (Pull) 100% (Push)	100% (Pull) 100% (Push)	100% (Pull) 100% (Push)
Simple RNN	51.96%	64% (B) 48.39% (K) 51.28% (Pull) 0% (Push) 28.57% (R)	80% (B) 68.18% (K) 100% (Pull) 0% (Push) 10% (R)	71% (B) 56.61% (K) 67.8% (Pull) 0% (Push) 14.82% (R)

Classification Results for Cyberbullying Detection on TikTok videos

We proposed a system for cyberbullying detection in TikTok videos using a deep-learning approach. Our solution employs a Transformer-based model that operates on Convolutional Neural Network

(CNN) feature maps. Table 6.4 describes the experiments we conducted with our deep learning-based model and the obtained accuracy. The *MAX_SEQ_LENGTH* field refers to the maximum number of frames we take from each video. When a video frame count is lesser than this field value, we pad the video with zeros. The *NUM_FEATURES* field represents the number of features the CNN model (i.e., DenseNet-121) extracts from each video. The *IMG_SIZE* field refers to the dimensions of the matrix that is cropped from the center of each frame. In our case, that will be 500x500 pixels.

Table 6.4. Classification results

Keras Application	MAX_SEQ_LENGTH	NUM_FEATURES	IMG_SIZE	EPOCHS	Accuracy
DenseNet-121	30	1024	500	20	Up to 100%

Some results we obtained during one of our experiments are presented in Table 6.5. They highlight the accuracy and loss values for each epoch for training and validation data. The *Loss* and *Accuracy* fields refer to training, while *Val_loss* and *Val_accuracy* are related to validation. We can say that the accuracy of the proposed solution is up to 100% since we got for epoch 14 an accuracy on validation data of 100%.

Table 6.5. Loss and accuracy values for each epoch

Epoch no.	Loss	Accuracy	Val_loss	Val_accuracy
1/20	3.4658	0.4251	8.8870	0.0000e+00
2/20	1.3992	0.6232	3.0966	0.0000e+00
3/20	0.5996	0.8213	2.0553	0.1081
4/20	0.2851	0.9058	0.9450	0.6081
5/20	0.2175	0.9251	2.0311	0.3514
6/20	0.2009	0.9469	1.6759	0.4730
7/20	0.1240	0.9638	0.2524	0.8649
8/20	0.0453	0.9831	3.1626	0.3108
9/20	0.0217	0.9928	0.2196	0.9189
10/20	0.0284	0.9879	1.2308	0.6351
11/20	0.0178	0.9952	0.6859	0.7432
12/20	0.0186	0.9976	0.7603	0.7568
13/20	0.0319	0.9928	0.7513	0.7297
14/20	0.0393	0.9903	0.0052	1.0000
15/20	0.1854	0.9444	3.9171	0.4595
16/20	0.0410	0.9831	0.6692	0.8243
17/20	0.0333	0.9879	1.1296	0.7162
18/20	0.0597	0.9734	1.4036	0.6892
19/20	0.0425	0.9831	1.0926	0.7568
20/20	0.0168	0.9928	2.6229	0.5811

When testing our solution against a video whose content is about shooting, we got the following remarkable results depicted in Table 6.6. Figure 6.3 shows one frame from that video.

Table 6.7 compares the accuracy of our proposed architecture with one of the systems in the literature we took from GitHub and improved.

Figure 6.3. Test video with bullying content (Shoot).

Table 6.6. Test video results (Shoot)

Class	Probability
Shoot	100.00%
Basketball	0.00%
Football	0.00%
Kick	0.00%
Playing guitar	0.00%
Playing cello	0.00%

Table 6.7. Comparison in terms of accuracy between our proposed solution and the initial one

	Accuracy
Initial GitHub solution	67.5%
Proposed and improved solution *	Up to 100%

6.4 Conclusions

In this chapter, we discuss two solutions for cyberbullying detection in multimedia files. The first deals with GIF files, while another processes TikTok videos.

Concerning the first system, we create a dataset of bullying GIFs. With the help of a web scrapping tool, we took GIFs related to bullying from the GIPHY platform and filtered them until the most relevant ones remained. The accuracy of the proposed system is 99%. It can classify GIFs into *bullying*, *Bodybuilding*, and *Water sports*. Moreover, our solution can further classify the files of the last two categories. The former can be further classified into *Pull Ups* or *Handstand Pushups*, while the latter into *Rowing* or *Kayaking*.

As regards the second system, we propose a novel cyberbullying detection solution for TikTok videos using a deep learning-based model. We create a dataset of videos from TikTok with both bullying and non-bullying content. We manually downloaded them with the help of the SnapTik platform and processed them using the Clipchamp website. Further, we create a system that employs a Transformer-based model for video classification, which operates on Convolutional Neural Network (CNN) feature maps. We evaluated our model against the created dataset, and we got an accuracy of up to 100%.

We can conclude that our solutions are one step forward in the research and development of security systems, especially for the mitigation of bullying attacks performed via GIFs and TikTok

videos.

Possible use cases that might benefit from our proposal include online bullying or harassment in public or private environments such as universities, hospitals, hotels, or corporate buildings. Our proposals add value to Internet users by providing advanced capabilities for bullying detection in GIFs and TikTok videos.

7 | Next Generation Agent-based Endpoint Detection and Response Systems for Cybersecurity Threats

In this chapter, we discuss a next-generation agent-based EDR system for cybersecurity threats (NextEDR). It is an innovative and interactive Cloud-Edge-Continuum Endpoint Detection and Response platform for protecting modern organizations from cybersecurity attacks. We design a Proof-of-Concept based on an interactive communication agent (ChatBot) solution for phishing detection in short URLs. Our solution is a mobile-centric multi-layer platform based on the Cloud-Edge-Continuum model.

This project we proposed was part of a Proof of Concept program (*Invest National University of Science and Technology POLITEHNICA Bucharest Proof of Concept (Invest PoC)* program), and we worked on developing a business plan.

The content of this chapter is based on the publication of the paper *NextEDR - Next Generation Agent-Based EDR Systems for Cybersecurity Threats* in *2024 32nd Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP)*. This chapter is structured as follows: in Section 7.1, we propose the *NextEDR* platform and architecture. Section 7.2 presents the obtained classification results. Finally, in Section 7.3, we draw our conclusions and present future work.

7.1 NextEDR Architecture

In this section, we present the architecture of the proposed solution. It is described in Figure 7.1.

Our proposal includes four layers: (i) the data sources layer, (ii) the Edge layer, (iii) the Cloud layer, and (iv) the Application layer. First, the data sources layer includes emails, SMSs, Instant messages (e.g., WhatsApp, Signal, etc), and social media (e.g., Facebook, Instagram, etc.) on mobile devices. The Edge layer extracts the short URL from the data sources, converts it to the original form, and sends it to the Cloud layer via the ChatBot interface. There, it is processed by using two approaches: (1) through threat intelligence platforms (e.g., VirusTotal and PhishTank) and (2) through ML classifiers (e.g., JRip, PART, J48, and Random Forest). In addition, we introduce, in the Cloud layer, a novel interactive conversational agent. The main functionalities behaviors of ChatBot are to check the URL and train the platform by reporting malicious URLs. The end-user makes use of the ChatBot service to send a URL for verification. The ChatBot has three behaviors: (a) positive, (b) negative, or (c) unknown. To facilitate accurate reporting, the *NextEDR* platform implements rewarding mechanisms. Moreover, in case of a positive result, the Cloud layer sends notifications through the alert service to all mobile devices in the platform. This fact ensures a low communication overhead in the system and low incident response time in case

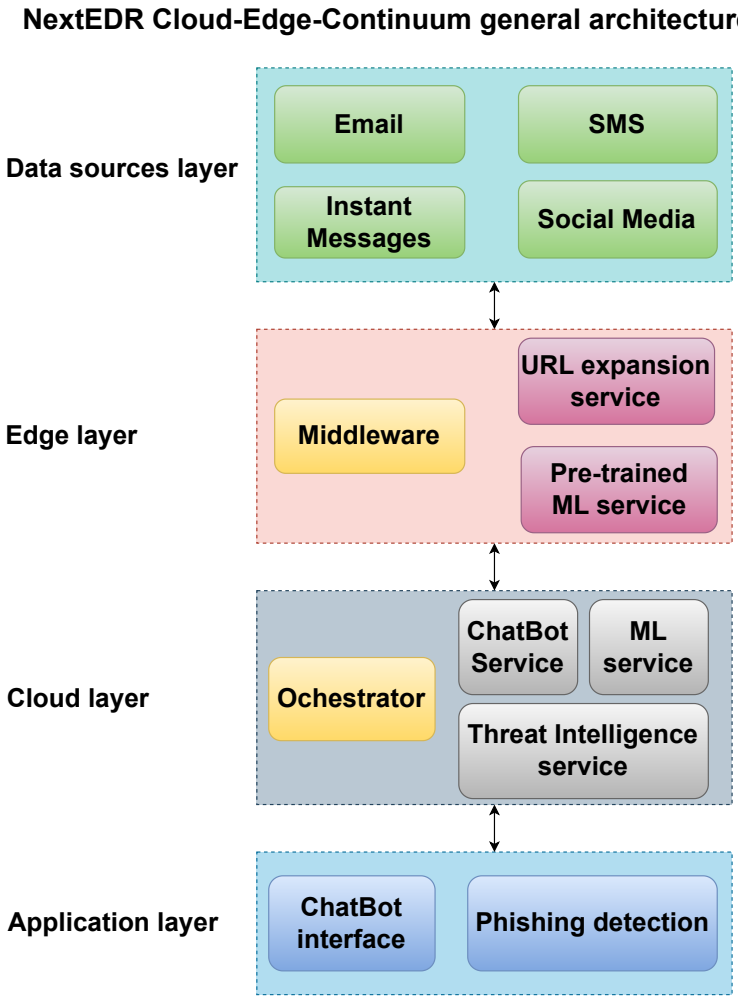

Figure 7.1. *NextEDR* general architecture.

of a phishing attack.

Always between the Edge and Cloud layers, where we have both the URL expansion and pre-trained ML services, and on the Orchestrator side, where we have the ML and Threat Intelligence services, we could bring better models that can change these two layers, while the Data sources and Application layers remaining the same. A use case would be the web browser where we would like to stop cyberbullying for children. The Data sources and Application layers remain unchanged while we can bring updates just for the Middleware and Orchestrator. It could be part of the business plan.

In future directions, our method will not involve changing the data sources or the application layer. The models will adapt to the data sources and application. This fact would represent an advantage.

7.2 Experimental Results and Analysis

In this section, we present the obtained results. Table 7.1 presents the analysis time of the threat intelligence approach for various short URLs.

Table 7.1. The analysis time of the Threat Intelligence approach.

Short URL	Final URL	T.I. time (s)
https://shorturl.at/fuxAE	https://call.raidstore.org/	81.267
https://t.ly/yEC	https://technology.macosevents.com/	2.394
https://rb.gy/w2i1u	https://press.infomapress.com/	1.889
https://rb.gy/wtkn	https://stiri.botosani.ro/	1.737
https://tinyurl.com/kyc44ft8	https://www.bbc.com/	2.717
https://cutt.ly/zwr2HIaS	https://www.amazon.com/b?node=21576558011 &ref_=alxcom_lrnmore_btn_23	2.864
https://urlis.net/b94wy861	https://sucursalcentroapp.brizy.site/	2.681
http://tiny.cc/tzb8vz	https://kuccoiieeiinelovuiie.godaddysites.com/	2.499
https://url1.io/s/VpEwo	https://anime-info777.com/	3.879

For the speed test of the combined feature extraction and ML classification processes, we chose ten URLs for each of the categories: phishes, legitimate, and unknown. Table 7.2 presents the results we obtained. It can be observed the time for ML classification varies between 2.4420 and 3.3585 seconds.

Table 7.2. Duration of ML classification.

Operation	Item	Duration for ML classification (s)
Testing VALID phishes	https://pub-c479da8c0e2748d0a34fd7266d91fc30.r2.dev/index.html	2.5128
	https://actividadbancaria–giovannyquint.repl.co/	2.4780
	https://highlight-himself.toshibanetcam.com/	2.4558
	https://dev-asistenonlibcr.pantheonsite.io/	2.5539
	https://homeless-hospital.otzo.com/	2.5590
	https://joint-knowledge.instanthq.com/	2.7674
	https://department-depict.mrface.com/	2.6300
	https://japanese-joint.qpoe.com/	2.5126
	https://approximately-arab.mrbasic.com/	2.5296
	https://administration-adopt.toythieves.com/	2.4420
Testing UNKNOWN phishes	https://ameli-france-connect.com/	2.6018
	https://portalemydati.online/	2.4354
	http://portalemydati.online/	2.5411
	http://infomydati.online/	2.6577
	https://pagamenti.staffasestenza.co/8328-1/	2.6222
	https://ckka7cxmhjcr6qbe.ipfs.dweb.link/	2.6082
	http://updatetan-sp.de/anmelden	2.7528
	https://www.gefhuloa.com/pl/pl_dfertz/?uclick=9lpmg9e712d	2.6611
	https://mail.kinepolis.com/optiext/optiextension.dll?ID=6yz6yKcj	2.7708
	https://khatampanjereh.com/wp-includes/knab.php	2.6577
Testing NOT phishes	https://www.cloudflare.com/	3.3585
	https://fonts.googleapis.com/css?family=Noticia+Text:400,400i,700,700i	2.6238
	https://express.adobe.com/	3.0263
	http://www.paypal-merchant.com/	2.5208
	https://www.mercadopago.com.br/	2.4097
	https://onedrive.live.com/about/es-us-signin/	2.5991
	http://url.zp.edu.ua/	2.5182
	https://bitflyer.com/en-us/	2.4980
	https://phishtank.org/phish_detail.php?phish_id=8153631	2.9479
	https://quttera.com/	2.4490

System Deployment

Four people worked on this project. It was me and my scientific supervisors of this thesis. I was the Delivery lead, while the others had functions such as Director and Academic mentor.

We launched a beta version of our EDR system to a carefully selected group of end-users representing each of our detailed personas. Throughout the beta testing phase, we collected detailed feedback through various methods including surveys, interviews, and direct user observations. We focused on understanding their experiences with the system, identifying any pain points, and gathering suggestions for improvement.

We closely monitored key performance metrics such as detection accuracy, response times, false-positive rates, and system usability. This data provided quantifiable evidence of the system's effectiveness and areas needing enhancement. Based on the feedback and performance metrics, we made iterative improvements to the system. This agile approach allowed us to quickly address any issue and refine the product to better meet user expectations.

According to the business model of the NextEDR project, potential customer segments for our system include enterprises, technology companies, industry verticals, educational institutions, and managed service providers. We could make our customers happy and satisfied with our solution by providing enhanced cybersecurity protection, customized and adaptive protection, a cost-effective and scalable solution, and a user-friendly interface and accessibility. Figure 7.2 describes the canvas business model for the NextEDR platform.

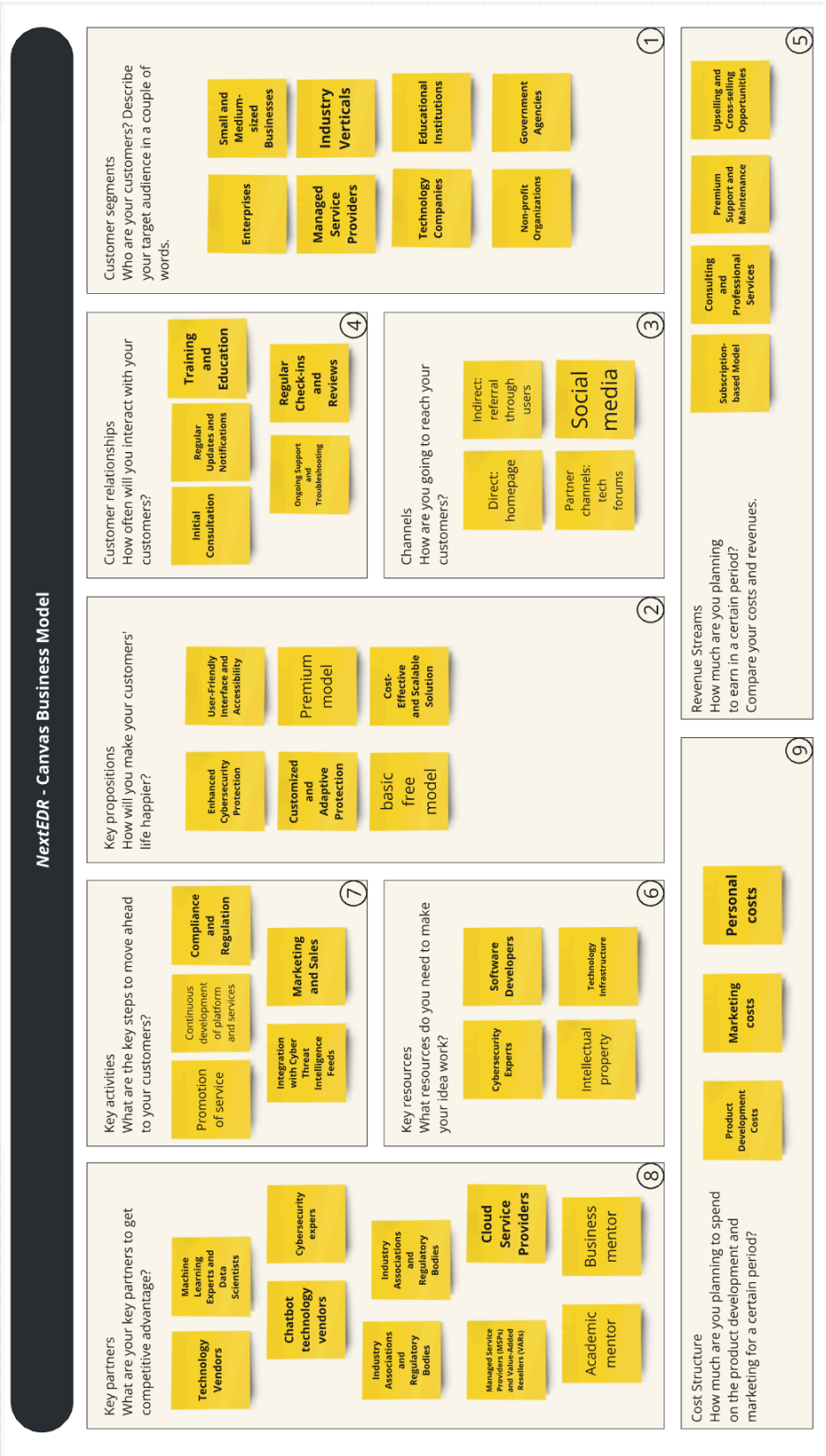

Figure 7.2. NextEDR - Canvas Business Model.

Real-case System Validation

We encountered a real-world situation when we received an SMS message on our smartphones that appeared sent by the Romanian post institution. The SMS message states the delivery address data for a parcel was lost, and this information needs to be updated by accessing the mentioned URL. The SMS message can be seen in Figure 7.3.

Figure 7.3. Suspicious SMS message received.

The URL in the SMS message looked suspicious, and we decided to check its legitimacy using the EDR system. Figure 7.4 presents the analysis results.

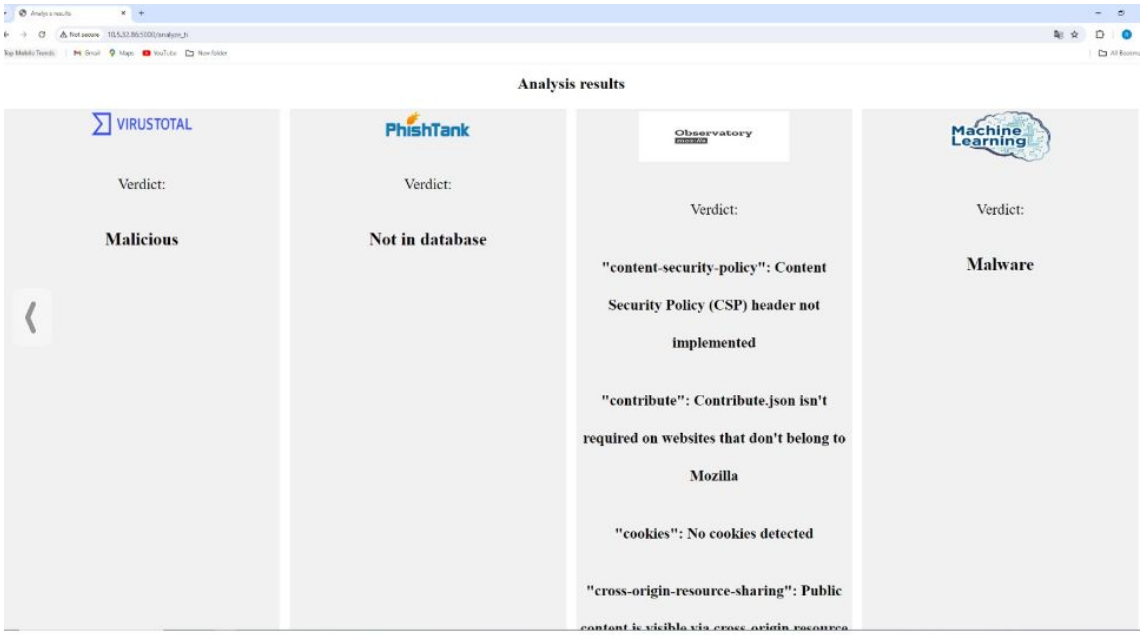

Figure 7.4. Suspicious SMS analysis using NextEDR.

It seems that VirusTotal and the Machine Learning model classified the URL as malicious.

PhishTank did not find the URL in the phishing reported URLs, maybe since the campaign was ongoing and very recent. The results generated by Mozilla Observatory reveal the security misconfigurations of the website the analyzed URL points to.

7.3 Conclusions

In this paper, we propose a next-generation Cloud-Edge-Continuum EDR platform that advances to a higher level of cybersecurity solutions through intelligent conversational agents called ChatBoots.

This solution earned the first prize at *Invest UNSTPB Proof of Concept* project competition. The Minimum Viable Product (MVP) development for the NextEDR platform involved iterative refinement based on user feedback, prioritizing essential features to deliver a functional solution that addresses core user needs efficiently.

Considering the design based on the Cloud-Edge-Continuum model, our solution reduces communication overhead and guarantees low latency and fast response time, as well as high availability through load-balancing technology. Our system is designed for smartphone devices and covers different scenarios: legitimate, suspicious, and malicious URLs that are shortened.

The results showed malicious short URLs can be identified, and the corresponding SMS is classified as smishing. Our approach has the advantage of using free and open-source tools that offer an efficient system for malicious short URL detection.

8 | Cybersecurity Governance Methodology in Large-scale Infrastructures

In this chapter, we discuss a framework for cybersecurity risk assessment in large-scale infrastructures alongside mitigation techniques and countermeasures. The methodology we employ in our study is qualitative. It is based on 66 projects from the CORDIS database of the European Commission related to cybersecurity governance in smart cities.

From the analysis of these projects, we extracted ideas that define cybersecurity governance and different prevention methodologies for cyber threats like cyberbullying, malicious short URLs, phishing attacks, etc.

The content of this chapter is based on the publication of the paper *Cybersecurity Governance in Large-scale Infrastructures* in *Romanian Journal of Information Technology and Automatic Control*.

This chapter is structured as follows: Section 8.1 presents an introduction to the field of the studied issue. In Section 8.2, we present our proposed methodology, while the main threats and potential countermeasures are described in Section 8.3. Finally, in Section 8.4, we conclude the results of the proposed framework and identify future research opportunities.

8.1 Background

Many cities around the world risk facing problems concerning life conditions since they have important issues regarding the security, scalability, and the environment of their infrastructures. It is due to the population growth that will reach 9.8 billion in 2050 [11]. As a result, the urban environment will encounter both challenges and benefits. Some of the difficulties it would face are represented by the fact that the education and the health sectors will need new approaches, the economy will have issues, the energy consumption will increase, public safety will face new risks, and the possibility of cyberattacks against cities is high. The key solution for these problems is innovative, scalable, and cost-effective infrastructures [5].

In this paper, we propose a model for cybersecurity risk assessment in large-scale infrastructures alongside mitigation techniques and countermeasures. The methodology we employ in our study is qualitative. It is based on 66 projects from the CORDIS database of the European Commission related to cybersecurity governance in smart cities. They are focused on research and innovation and belong to the date range between 2022 and 2027. Our work brings a significant contribution to the scientific community as we identify the security risks in large-scale infrastructure and propose mitigation techniques and countermeasures for these challenges.

8.2 Cybersecurity Governance Methodology

In this section, we present the methodology we proposed for cybersecurity risk assessment in large-scale infrastructures alongside mitigation techniques and countermeasures.

To identify the ongoing development trends for the smart era we live in, we began our study by evaluating research projects of the European Commission. They are stored in the CORDIS database and belong to different fields of activity, such as medicine, transport, buildings, education, and technology. They are proposed and implemented by universities, research centers, and corporates. By filtering the projects by the two most important themes of our paper, "cybersecurity" and "infrastructure," we got 66 results. Then, we took and imported each project into NVivo 14, a tool for qualitative data analysis. We assigned a theme to each of the projects and grouped the ones of the same type together so that we could identify which are the most prevalent sectors of activity or areas of large-scale infrastructure. Based on them, we spot where security risks could appear and which could be the potential mitigation techniques and countermeasures. Table 8.1 briefly presents the steps we adopted in our methodology.

Table 8.1. The proposed methodology.

Phase	Description
1. Data gathering	We searched for research projects of the European Commission in the CORDIS database.
2. Data filtering	We filtered the projects by the "cybersecurity" and "infrastructure" keywords.
3. Project import	We took each obtained project and imported it into NVivo 14, a tool for qualitative data analysis.
4. Theme assigning	We analyzed each project and assigned a theme.
5. Theme grouping	We grouped the themes of the same type and identified the main areas they belong to.
6. Security risks	We analyzed each identified area from a large-scale infrastructure and assigned security risks.
7. Mitigation	We proposed mitigation techniques and countermeasures for the identified security risks.

8.3 Main Threats and Potential Countermeasures

In this section, we present the main threats and potential countermeasures against the areas in a large-scale infrastructure that we identified through the proposed cybersecurity risk assessment framework.

Once we obtained all the projects from the CORDIS database, we imported them into NVivo 14, a tool for qualitative data analysis. There, we processed every project and assigned a theme based on its objectives. The themes corresponding to a specific area in a large-scale infrastructure were grouped.

We present in Table 8.2 the security threats and potential countermeasures for each area in a large-scale infrastructure that we identified through the proposed cybersecurity risk assessment framework. We specified in the table the number of projects from the CORDIS database we selected for each domain. The threats column refers to the main cyber-security challenges that target a specific area. We have reviewed multiple research papers from the literature to define them.

The countermeasures column proposes actions that could be taken to defend and prevent cyber attacks. It also contains, for each item, the number of projects it covers. The value represents the sum of all projects that implement or should employ that specific countermeasure. One project can need or contain more countermeasures.

The coverage column displays a mapping between projects and the number of countermeasures they either implement or should employ.

Table 8.2. The main threats and potential countermeasures.

Area	Threats	Countermeasures	Coverage
Technology 16 projects	- Unauthorized access - Man-in-the-middle attacks - Stole research & development data - Vulnerability exploitation	- Vulnerability assessment – 10 projects - Authentication and authorization mechanisms – 6 projects - Intrusion detection systems – 10 projects - Data Loss Prevention systems – 5 projects	- 1 project approaches 4 countermeasures - 2 projects approach 3 countermeasures - 8 projects approach 2 countermeasures - 5 projects approach 1 countermeasure
Security 12 projects	- Code vulnerabilities - Out of date database of malware signatures - Killing the anti-virus (AV) process	- Make a more secure code – 7 projects - Keep an up-to-date database of malware signatures – 0 projects - Prevent the AV process from being killed by malicious software – 0 projects - Threat Intelligence data integration – 6 projects - Disaster recovery/business continuity plan – 7 projects	- 1 project approaches 3 countermeasures - 6 projects approach 2 countermeasures - 5 projects approach 1 countermeasure
Healthcare 9 projects	- Sensitive data exposure - Disrupting the services - Eavesdropping sensitive information - Sending fake information - Patients' data alteration	- Secured Wi-Fi networks to guarantee safe handling of confidential information and personal data (e.g., AirTight Networks solutions) – 6 projects - Risk assessment (e.g., Intel healthcare security solutions) – 4 projects	- 2 projects approach 2 countermeasures - 7 projects approach 1 countermeasure
Energy 8 projects	- Unauthorized access and controls - Botnets (e.g., Zeus, Conficker) - Denial of service (DoS) and distributed denial of service (DDoS) attacks	- Intrusion detection and prevention systems (e.g., Snort) – 3 projects - Cyber Threat Intelligence – 2 projects - Risk assessment methodologies (e.g., MEHARI, EBIOS) – 8 projects	- 2 projects approach 3 countermeasures - 1 project approaches 2 countermeasures - 5 projects approach 1 countermeasure
Environment 7 projects	- Attacks against the network and PLCs - System compromise - Vulnerabilities	- Vulnerability patching – 2 projects - Security monitoring solutions – 6 projects - Water Information Sharing and Analysis Center, American Water Works Association – 1 project	- 2 projects approach 2 countermeasures - 5 projects approach 1 countermeasure
Infrastructure 4 projects	- Supply chain attacks - Insecure communication - Weak authentication	- Asset inventory and risk assessment – 3 projects - Security patching and updates – 3 projects - Supply chain security – 3 projects	- 2 projects approach 3 countermeasures - 1 project approaches 2 countermeasures - 1 project approaches 1 countermeasure
Transportation 4 projects	- Braking system disruption - Engine stopping - Displaying false messages at the on-board computer - Changing GPS signals	- The use of cryptography (digital certificates, Public key infrastructure, data encryption) – 3 projects - Solutions for anomaly detection – 1 project	- 1 project approaches 2 countermeasures - 2 projects approach 1 countermeasure - 1 project (i.e., NextETRUCK) does not cover any of these main countermeasures
Education 3 projects	- Data breaches - Personal information compromise - Ransomware attacks	- Anti-malware and anti-virus software – 0 projects - Using strong passwords – 0 projects - Security awareness training – 3 projects	- 3 projects approach 1 countermeasure
Citizen 2 projects	- Cybercrime - Identity theft	- Awareness training – 2 projects - Use of strong passwords – 0 projects	- 2 projects approach 1 countermeasure
Governance 1 project	- Disrupting critical infrastructures - Fiscal fraud - Altered files	- Data Loss Prevention solutions (e.g., Symantec, Fortinet) – 0 projects - Risk assessment methodologies (e.g., MEHARI, EBIOS) – 1 project - Insider threat analysis – 0 projects	- 1 project approaches 1 countermeasure

8.4 Conclusions

In this paper, we propose a model for cybersecurity risk assessment in large-scale infrastructures alongside mitigation techniques and countermeasures. We employ in our study a qualitative methodology by identifying 66 European research projects from the period 2022 to 2027 that are related to cybersecurity governance in large-scale infrastructures. We import them into the NVivo tool for qualitative data analysis. There, we group them by the sector of activity so that we can identify the most prevalent areas. The latter are split into the main components based on which we identify the cybersecurity threats.

The results of our work offer significant scientific contributions by identifying security risks in large-scale infrastructure and proposing mitigation techniques and countermeasures for these challenges. The obtained results are limited since we have searched for research projects in the CORDIS database just by the "cybersecurity" and "infrastructure" keywords. If we had added other terms in the search, such as "smart city" and "governance," maybe we would have obtained more results. Thus, the domains of activity would be larger, and the range of identified threats and proposed countermeasures would be bigger. In this way, our study analyzes some areas and might not treat all the domains that large-scale infrastructures are compound of.

In terms of future work, we intend to enrich the current model by presenting some practical use cases where we can describe real data breaches. It would be useful since we can come up with concrete and specific intrusion detection tactics and countermeasures.

9 | Conclusions and Future Directions

This thesis focuses on the identification of the most prevalent cyber attack types in the modern era and the development of detection strategies, countermeasures procedures, and robust security solutions that provide efficiency, accurate results, and fast response time.

9.1 Original Contributions

1. We have identified the modern cyber security attacks that describe the threat landscape for the 2021-2022 period.
2. We proposed some contemporary countermeasures and detection strategies that help to defend against the most prevalent cyber threats.
3. We proposed a malicious short URL detection technique.
4. We proposed a scalable malicious URL detection technique for Smishing attacks.
5. We proposed a bullying detection solution for GIFs using a deep learning approach.
6. We proposed a solution for cyberbullying detection on TikTok using a deep learning approach.
7. We proposed NextEDR - Next generation agent-based EDR systems for cybersecurity threats.
8. We proposed a framework for cybersecurity governance in large-scale infrastructures.
9. I have elaborated on prevention methodologies for the threats tackled by the detection-based security solutions we proposed.

9.2 Publications and Projects

The main results of this thesis were presented at various conferences and journals. I have seven publications, six as first author and one as co-author. My publications list consists of 4 articles in international journals (International Journal of Computational Science and Engineering, Information, UPB Sci. Bull., Series C, and Romanian Journal of Information Technology and Automatic Control) and three papers in well-established international conferences (2023 24th International Conference on Control Systems and Computer Science (CSCS), 2022 21st RoEduNet Conference: Networking in Education and Research (RoEduNet), and 2024 32nd Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP)).

I want to thank the reviewers for their time, expertise, constructive comments, and valuable insight.

Previously published papers:

1. **Stoleriu, Razvan**; Negru, Catalin; Radulescu, Dragos; "Modern Cyber Security Attacks, Detection Strategies, and Countermeasures Procedures," "2023 24th International Conference on Control Systems and Computer Science (CSCS)," pp. 198-205, 2023, IEEE, doi: 10.1109/CSCS59211.2023.00039.

2. **Stoleriu, Razvan**; Negru, Catalin; Mocanu, Bogdan-Costel; Pop, Florin; "Malicious Short URLs Detection Technique," "2023 22nd RoEduNet Conference: Networking in Education and Research (RoEduNet)," pp. 1-6, 2023, IEEE, doi: 10.1109/RoEduNet60162.2023.10274913.

3. **Stoleriu, Razvan**; Negru, Catalin; Mocanu, Bogdan-Costel; Constantinescu, Emil-Andrei; Mocanu, Alexandra-Elena; Pop, Florin; "Scalable Malicious URL Detection Technique for Smishing Attacks," "International Journal of Computational Science and Engineering," Inderscience Publishers (IEL).

4. Mocanu, Bogdan-Costel; **Stoleriu, Razvan**; Mocanu, Alexandra-Elena; Negru, Catalin; Dragotoiu, Elena-Gabriela; Moisescu, Mihnea-Alexandru; Pop, Florin; "NextEDR-Next generation agent-based EDR systems for cybersecurity threats," "2024 32nd Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP)," pp. 183-190, 2024, IEEE, doi: 10.1109/PDP62718.2024.00033.

5. **Stoleriu, Razvan**; Nascu, Andrei; Anghel, Ana Magdalena; Pop, Florin; "Bullying Detection Solution for GIFs Using a Deep Learning Approach," "Information," 15, 2024, MDPI, doi: https://www.mdpi.com/2078-2489/15/8/446.

6. **Stoleriu, Razvan**; Nascu, Andrei; Pop, Florin; "Cyberbullying Detection on TikTok Using a Deep Learning Approach," "UPB Sci. Bull., Series C", 2024. - **Accepted paper**

7. **Stoleriu, Razvan**; Petre, Ionut; Pop, Florin; "Cybersecurity Governance in Large-scale Infrastructures," "Romanian Journal of Information Technology and Automatic Control," 2025. - **Accepted paper (ISI)**

During the PhD period, I was a member of a project for which I am thankful, offering me the context to build real-world use cases for my thesis and the opportunity to interact with different researchers. The project I was part of earned the first prize at *Invest UNSTPB Proof of Concept* project competition. This project is:

- INVEST UPB Proof of Concept – Phase 2, for project NextEDR - Next generation agent-based EDR systems for cybersecurity threats, funded by project DECIP: Dezvoltarea Capacității Instituționale a Universității POLITEHNICA din București, Contract PFE 22, Period: 24 November – 10 June 2024 (7 months), Director: Asst. Bogdan-Costel Mocanu.

Bibliography

[1] Asmaa Shaker Ashoor and Sharad Gore. Difference between intrusion detection system (ids) and intrusion prevention system (ips). In *Advances in Network Security and Applications: 4th International Conference, CNSA 2011, Chennai, India, July 15-17, 2011 4*, pages 497–501. Springer, 2011.

[2] Andreea Bendovschi. Cyber-attacks–trends, patterns and security countermeasures. *Procedia Economics and Finance*, 28:24–31, 2015.

[3] Sarika Choudhary, Ritika Saroha, and Mrs Sonal Beniwal. How anti-virus software works?? *International Journal*, 3(4):483–484, 2013.

[4] Prithviraj Dasgupta, Joseph B Collins, and Ranjeev Mittu. *Adversary-Aware Learning Techniques and Trends in Cybersecurity*. Springer, 2021.

[5] Rida Khatoun and Sherali Zeadally. Cybersecurity and privacy solutions in smart cities. *IEEE Communications Magazine*, 55(3):51–59, 2017.

[6] Sean Lawson. Beyond cyber-doom: Cyberattack scenarios and the evidence of history. *Mercatus Center at George Mason University*, 2011.

[7] Richard Lippmann, Seth Webster, and Douglas Stetson. The effect of identifying vulnerabilities and patching software on the utility of network intrusion detection. In *Recent Advances in Intrusion Detection: 5th International Symposium, RAID 2002 Zurich, Switzerland, October 16–18, 2002 Proceedings 5*, pages 307–326. Springer, 2002.

[8] Michael R Lyu and Lorrien KY Lau. Firewall security: Policies, testing and performance evaluation. In *Proceedings 24th Annual International Computer Software and Applications Conference. COMPSAC2000*, pages 116–121. IEEE, 2000.

[9] Adam McNeil and W Stuart Jones. Mobile malware is surging in europe: A look at the biggest threats, 2022.

[10] Bogdan-Costel Mocanu, Razvan Stoleriu, Alexandra-Elena Mocanu, Cătălin Negru, Elena-Gabriela Drăgotoiu, Mihnea-Alexandru Moisescu, and Florin Pop. Nextedr-next generation agent-based edr systems for cybersecurity threats. In *2024 32nd Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP)*, pages 183–190. IEEE, 2024.

[11] United Nations. World population projected to reach 9.8 billion in 2050, and 11.2 billion in 2100 | united nations. https://www.un.org/en/desa/world-population-projected-reach-98-billion-2050-and-112-billion-2100, 2017. [Online; accessed 18-January-2024].

[12] CHECK POINT. 2022 cyber attacks trends: Mid-year report. https://resources.checkpoint.com, 2022. [Online; accessed 06-October-2023].

[13] Jamal Raiyn et al. A survey of cyber attack detection strategies. *International Journal of Security and Its Applications*, 8(1):247–256, 2014.

[14] Jungwoo Ryoo, Syed Rizvi, William Aiken, and John Kissell. Cloud security auditing: challenges and emerging approaches. *IEEE Security & Privacy*, 12(6):68–74, 2013.

[15] IBM Security. X-force threat intelligence index 2022. https://www.ibm.com/downloads/cas/ADLMYLAZ, 2022. [Online; accessed 10-April-2024].

[16] A Seetharaman, Nitin Patwa, Veena Jadhav, AS Saravanan, and Dhivya Sangeeth. Impact of factors influencing cyber threats on autonomous vehicles. *Applied Artificial Intelligence*, 35(2):105–132, 2021.

[17] Ioannis Stellios, Kostas Mokos, and Panayiotis Kotzanikolaou. Assessing smart light enabled cyber-physical attack paths on urban infrastructures and services. *Connection Science*, 34(1):1401–1429, 2022.

[18] Razvan Stoleriu, Andrei Nascu, Ana Magdalena Anghel, and Florin Pop. Bullying detection solution for gifs using a deep learning approach. *Information*, 15(8):446, 2024.

[19] Razvan Stoleriu, Catalin Negru, Bogdan-Costel Mocanu, and Florin Pop. Malicious short urls detection technique. In *2023 22nd RoEduNet Conference: Networking in Education and Research (RoEduNet)*, pages 1–6. IEEE, 2023.

[20] Răzvan Stoleriu, Cătălin Negru, and Dragos Rădulescu. Modern cyber security attacks, detection strategies, and countermeasures procedures. In *2023 24th International Conference on Control Systems and Computer Science (CSCS)*, pages 198–205. IEEE, 2023.

[21] Kami Vaniea and Yasmeen Rashidi. Tales of software updates: The process of updating software. In *Proceedings of the 2016 chi conference on human factors in computing systems*, pages 3215–3226, 2016.

[22] Zhensheng Zhang, Ya-Qin Zhang, Xiaowen Chu, and Bo Li. An overview of virtual private network (vpn): Ip vpn and optical vpn. *Photonic network communications*, 7:213–225, 2004.

[23] Aaron Zimba. Malware-free intrusion: a novel approach to ransomware infection vectors. *International Journal of Computer Science and Information Security*, 15(2):317, 2017.

[24] AlMaha Abu Zuraiq and Mouhammd Alkasassbeh. Phishing detection approaches. In *2019 2nd International Conference on new Trends in Computing Sciences (ICTCS)*, pages 1–6. IEEE, 2019.