Universidade Estadual de Londrina

RILDO ANTONIO DE SOUZA

A METHODOLOGY FOR MALWARE INCIDENT PREDICTION IN HIGHER EDUCATION INSTITUTIONS

LONDRINA

2024

RILDO ANTONIO DE SOUZA

A METHODOLOGY FOR MALWARE INCIDENT PREDICTION IN HIGHER EDUCATION INSTITUTIONS

Dissertação apresentada ao Programa de Mestrado em Ciência da Computação da Universidade Estadual de Londrina para obtenção do título de Mestre em Ciência da Computação.

Orientador: Prof. Dr. Bruno Bogaz Zarpelão

LONDRINA
2024

Ficha de identificação da obra elaborada pelo autor, através do Programa de Geração Automática do Sistema de Bibliotecas da UEL

D278a de Souza, Rildo Antonio.
A METHODOLOGY FOR MALWARE INCIDENT PREDICTION IN HIGHER EDUCATION INSTITUTIONS / Rildo Antonio de Souza. - Londrina, 2024.
72 f. : il.

Orientador: Bruno Bogaz Zarpelão.
Dissertação (Mestrado em Ciência da Computação) - Universidade Estadual de Londrina, Centro de Ciências Exatas, Programa de Pós-Graduação em Ciência da Computação, 2024.
Inclui bibliografia.

1. Malware - Tese. 2. Cybersecurity Incidents - Tese. 3. Incident Forecasting - Tese. 4. Higher Education Institutions - Tese. I. Zarpelão, Bruno Bogaz. II. Universidade Estadual de Londrina. Centro de Ciências Exatas. Programa de Pós-Graduação em Ciência da Computação. III. Título.

CDU 519

RILDO ANTONIO DE SOUZA

A METHODOLOGY FOR MALWARE INCIDENT PREDICTION IN HIGHER EDUCATION INSTITUTIONS

Dissertação apresentada ao Programa de Mestrado em Ciência da Computação da Universidade Estadual de Londrina para obtenção do título de Mestre em Ciência da Computação.

BANCA EXAMINADORA

Orientador: Prof. Dr. Bruno Bogaz Zarpelão
Universidade Estadual de Londrina

Prof. Dr. Rodolfo Miranda de Barros
Universidade Estadual de Londrina

Prof. Dr. Silvio Ereno Quincozes
Universidade Federal do Pampa

Londrina, 09 de Dezembro de 2024.

Este trabalho é dedicado aos navegantes das estrelas, que enxergam no céu não um limite, mas um convite para descobrir o infinito.

AGRADECIMENTOS

Agradeço, primeiramente, a Deus, pela força e orientação que me sustentaram em cada passo dessa jornada.

À minha esposa e à minha filha, por iluminarem minha vida com amor, paciência e compreensão. Sem o apoio e o carinho de vocês, este sonho não teria se tornado realidade.

À minha mãe e aos meus irmãos, que sempre acreditaram em mim, me incentivaram a seguir em frente e estiveram ao meu lado em cada etapa da minha trajetória.

Ao meu orientador, professor Bruno Bogaz Zarpelão, expresso minha profunda gratidão não apenas pela orientação técnica e acadêmica, mas também pelo apoio constante, pelos incentivos, pelos conselhos preciosos e pela paciência ao longo de todo o processo. Sua dedicação foi fundamental para a realização deste trabalho.

Aos professores Rodolfo Miranda e Silvio Ereno, agradeço imensamente por aceitarem o convite para compor a banca examinadora. Suas contribuições e sugestões enriqueceram significativamente esta dissertação.

Por fim, gostaria de agradecer aos meus colegas de trabalho, que tanto me apoiaram neste projeto. Em especial, ao Marcos Rodrigues e à Estefânia, minha gratidão pela ajuda e suporte nos momentos mais desafiadores. O apoio de vocês foi essencial para que eu conseguisse alcançar este objetivo.

"Embora ninguém possa voltar atrás e fazer um novo começo, qualquer um pode começar agora e fazer um novo fim."

— Chico Xavier

RESUMO

Com a crescente ameaça dos malwares afetando organizações globais, públicas e privadas, torna-se necessário desenvolver métodos inovadores para melhorar a prevenção e resposta a incidentes de segurança deste tipo. Infecções por *trojans*, *worms* que se propagam rapidamente, e *ransomware* que criptografa dados críticos, exigem respostas robustas devido ao risco que representam à integridade e disponibilidade das informações. Este trabalho apresenta uma metodologia que utiliza dados históricos de incidentes de malware para auxiliar CSIRTs (*Computer Security Incident Response Teams*) e equipes de segurança na análise de tendências comportamentais. A essência do método é um modelo de rede neural que emprega LSTM (*Long Short-Term Memory*) para análise de séries temporais, visando prever futuros incidentes de malware. O modelo foi testado em duas situações distintas em Instituições de Ensino Superior Públicas (IES) associadas à Rede Nacional de Ensino e Pesquisa (RNP). No primeiro cenário, utilizaram-se dados semanais de 10 instituições para treinar a LSTM, comparando sua eficácia com a técnica estatística ARIMA (Autoregressive Integrated Moving Average). No segundo cenário, expandiu-se a análise para incluir todas as IES clientes da RNP, segmentadas pelos Pontos de Presença (PoPs) presentes em cada estado brasileiro, demonstrando a aplicabilidade e versatilidade do método em um contexto ampliado. Para o desenvolvimento e a avaliação da metodologia, empregaram-se dados reais de incidentes, permitindo uma análise detalhada do desempenho dos modelos LSTM e ARIMA.Os resultados obtidos comprovam a eficácia do método proposto na previsão de incidentes de malware, destacando a criação e implementação de um modelo LSTM como a atividade central. Além disso, o método incorpora uma ferramenta de visualização de dados que simplifica a análise e interpretação das informações, facilitando o trabalho dos analistas de segurança ao traduzir as previsões em insights claros e acionáveis. Essa combinação entre precisão preditiva e acessibilidade analítica fortalece significativamente as estratégias de prevenção e resposta a incidentes. Como parte do compromisso com a comunidade, o método desenvolvido, bem como as ferramentas associadas, serão disponibilizados publicamente, visando apoiar o desenvolvimento de novas soluções de segurança, melhorar a detecção proativa de ameaças e contribuir para um ecossistema de segurança digital mais robusto.

Palavras-chave: Malware; Incidentes de cibersegurança; Previsão de incidentes; Instituições de Ensino Superior (IES); Long Short-Term Memory (LSTM) Network; Time Series; Autoregressive Integrated Moving Average (ARIMA).

SOUZA, R. A. **A METHODOLOGY FOR MALWARE INCIDENT PREDICTION IN HIGHER EDUCATION INSTITUTIONS**. 2024. 73p. Master's Thesis (Master in Science in Computer Science) – State University of Londrina, Londrina, 2024.

ABSTRACT

With the growing threat of malware affecting global organizations, both public and private, it becomes necessary to develop innovative methods to improve the prevention and response to such security incidents. Infections caused by *trojans*, rapidly propagating *worms*, and *ransomware* that encrypts critical data demand robust responses due to the risks they pose to the integrity and availability of information. This work presents a methodology that leverages historical malware incident data to assist CSIRTs (*Computer Security Incident Response Teams*) and security teams in analyzing behavioral trends. The core of the method is a neural network model employing LSTM (*Long Short-Term Memory*) for time series analysis, aiming to predict future malware incidents. The model was tested in two distinct scenarios in Public Higher Education Institutions (HEIs) associated with the National Research and Education Network (RNP). In the first scenario, weekly data from 10 institutions were used to train the LSTM, comparing its effectiveness with the statistical ARIMA (Autoregressive Integrated Moving Average) technique. In the second scenario, the analysis was expanded to include all HEIs that are RNP clients, segmented by the Points of Presence (PoPs) in each Brazilian state, demonstrating the method's applicability and versatility in an extended context. For the development and evaluation of the methodology, real incident data were employed, enabling a detailed analysis of the performance of the LSTM and ARIMA models. The results obtained confirm the effectiveness of the proposed method in predicting malware incidents, emphasizing the creation and implementation of an LSTM model as the core activity. Moreover, the method incorporates a data visualization tool that simplifies the analysis and interpretation of information, facilitating the work of security analysts by translating predictions into clear and actionable insights. This combination of predictive accuracy and analytical accessibility significantly strengthens prevention and response strategies for incidents. As part of the commitment to the community, the developed method, along with the associated tools, will be made publicly available, aiming to support the development of new security solutions, improve proactive threat detection, and contribute to a more robust digital security ecosystem.

Keywords: Malware; Cybersecurity Incidents; Incident Forecasting, Higher Education Institutions (HEIs); Long Short-Term Memory (LSTM) Network; Time Series; Autoregressive Integrated Moving Average (ARIMA).

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

ADAM	Adaptive Moment Estimation
AI	Artificial Intelligence
ARIMA	Autoregressive Integrated Moving Average
BPM	Business Process Management
BRNN-LSTM	Bidirectional Recurrent Neural Networks
CAIS-RNP	Security Incident Response Center for RNP
CERT.br	Center for Study, Response, and Treatment of Security Incidents in Brazil
CERTs	Computer Emergency Response Teams
CIRCL	Computer Incident Response Center Luxembourg
COBIT	Control Objectives for Information and Related Technology
CSIRT	Computer Security Incident Response Teams
CTIR.Gov	Government Cyber Incident Prevention, Response and Treatment Center
CVE	Common Vulnerabilities and Exposures
ENISA	European Union Agency for Cybersecurity
ETIR	Cybersecurity Response Team
FIRST	Forum of Incident Response and Security Teams
GRU	Gated Recurrent Units
HEIs	Higher Education Institutions
IEC	International Electrotechnical Commission
IETF	Internet Engineering Task Force
IODEF	Incident Object Description Exchange Format
IPS	Intrusion Prevention System

ISO	International Organization for Standardization
ITIL	Information Technology Infrastructure Library
LSTM	Long Short-Term Memory
KNN	K-Nearest Neighbors
MAE	Mean Absolute Error
ML	Machine Learning
MSE	Mean Squared Error
NCSCs	National Cybersecurity Centers
NIST	National Institute of Standards and Technology
PoP	Point of Presence
ReLU	Rectified Linear Unit
RFC	Request for Comments
RNN	Recurrent neural networks
RNP	Brazilian National Research and Education Network
X-ARF	Extended Abuse Reporting Format
SIEM	Security Information and Event Management
SIM3	Security Incident Management Maturity Model
SIRTs	Security incident response teams
STIX	Structured Threat Information eXpression
TimeGAN	Time-series Generative Adversarial Networks

CONTENTS

1 INTRODUCTION

A computer security incident is a violation or imminent threat of violation of computer security policies, acceptable use policies, or standard security practices [1]. Among various types, this work focuses on malware incidents, which represent a continual threat to cybersecurity. Malware infiltrates systems and devices to steal data, disrupt services, and cause financial harm. The frequency of malware incidents is on the rise, as reported by [2].

A brief examination of the malware campaign landscape in the past few years provides insight into the diverse threats and targets involved in these incidents. In 2019, notable examples included Ryuk, Emotet, and Trickbot. Ryuk was particularly involved in U.S. healthcare sector attacks [3]. In 2020, Emotet and Trickbot continued to be active, while new threats like Conti and Maze appeared. Conti targeted Ireland's Health Service Executive [4], and Maze attacked various international organizations, including Cognizant, a Fortune 500 IT company [5].

2021 saw the emergence of Hive, Lockbit, and REvil as new threat actors. REvil executed a notable attack on the Colonial Pipeline in the USA [6]. Hive was used against Costa Rica's Government and Zambia's Bank [7]. Lockbit 2.0, known for its advanced exploitation techniques, posed a significant concern [8]. That year, the Apache Log4j vulnerability (CVE-2021-44228), identified as a Common Vulnerabilities and Exposures (CVE) entry, emerged as a critical issue in November. In 2022, the malware landscape remained largely unchanged, with BlackCat added to the list [9]. The exploitation patterns were similar to the previous year, including continued attacks on Log4j and Windows OS, plus new vulnerabilities in VMware software [10].

The education and research industries account for a significant part of these incidents, as indicated in a Check Point's report [11]. Educational institutions present a concerning combination of attractive resources for attackers and difficulties in keeping a secure digital environment. Schools and universities host large amounts of personal and research data, making them highly profitable targets for cyber threats. Additionally, these organizations have become increasingly dependent on digital technologies for teaching, research and administrative activities. Cyber threats that disrupt these services may cause significant damage. Protecting these environments is particularly hard due to their diverse user base (students, teachers, researchers, and admin staff) who may operate within the institution premises or remotely. Universities, in particular, are naturally open environments, where the use of personal devices is often encouraged. On top of that, these organizations struggle to keep up-to-date security controls as they typically rely on public funds and operate within constrained budgets [12].

Organizations characterized by user diversity, extensive asset portfolios and budget constraints require tools that provide security analysts with a comprehensive perspective on the current scenario and improve their ability to utilize available resources more efficiently. Forecasting malware incident rates emerges as a valuable tool that helps to meet these needs. By knowing the expected rates of incidents, security analysts can anticipate trends, quickly detect outbreaks, and identify which assets are more or less prone to get infected. Additionally, the institution's IT (Information Technology) governance may see a major boost from the application of this method. By using the method, IT governance is strengthened through factors such as risk mitigation, efficient resource utilization, and the ability to foresee and promptly respond to incidents. This approach not only ensures service continuity but also supports stronger management practices aligned with governance principles outlined in frameworks such as Control Objectives for Information and Related Technologies (COBIT) [13] and the Information Technology Infrastructure Library (ITIL) [14]. Furthermore, this management data can also be exchanged with other organizations through specialized bodies like CSIRTs (Computer Security Incident Response Team), consequently improving overall cybersecurity conditions in the long term.

The objective of this work is to develop a method that enables CSIRTs or security teams to perform trend and behavioral variation analyses based on predictions derived from the security incident history in their respective areas of operation. The proposed methodology begins with the construction and validation of a neural network model, which is the core activity of the method, aimed at ensuring its ability to generate accurate projections. In this context, time series analysis of historical data is used, applying the concept of LSTM (Long-Short Term Memory) to achieve precise estimates. The process continues with the implementation of a script programmed to extract malware incidents from the incident data repository. After validating the data, it is cataloged in a database prepared for this purpose. The next step involves executing the previously established model, aimed at forecasting incident rates, with these projections properly recorded for future consultations and analyses. A crucial step in the methodology is the use of a visualization tool that allows security analysts to carry out a detailed inspection and interpretation of the information related to incidents and their respective forecasts. Once processed and made available, security analysts use this information in an analytical and decision-making approach, thus promoting a proactive attitude.

In situations where the forecasts prove to be non-assertive, the prediction model is adjusted to refine its accuracy and effectiveness. This work considers the process complete only after the successful completion of all stages, culminating in an optimized operational method that significantly contributes to the early identification of anomalous patterns and the emergence of new malware campaigns, thereby aiding in the prevention and response to subsequent security incidents. Additionally, the research aims to explore various pertinent questions related to the application of LSTM in predicting malware incidents: What

is the predictive efficacy of LSTM in these scenarios? How does the size of the institution influence the error rate in predictions? Can increasing the number of weeks provided as input to the model decrease prediction errors? Can more accurate predictions be obtained by extending the model's training period? Is an LSTM-based model more accurate than one based on the Autoregressive Integrated Moving Average (ARIMA) model?

To evaluate the proposed method, data from the RNP (Brazilian National Research and Education Network) incident records were used. These data include approximately 100,000 malware incidents in Brazilian higher education institutions over four years. For a detailed analysis, the data were divided into two distinct scenarios and organized into time series for each institution analyzed. The LSTM model was trained to predict the number of occurrences in the immediately following week, based on the input period.

The manuscript is structured as follows: Chapter 2 addresses the theoretical foundation, providing essential definitions such as information security incidents, IT governance and management, malware, and the role of a CSIRT. Additionally, concepts related to machine learning, time series analysis, and recurrent neural networks are explored, concluding with a review of related work in the area of security event forecasting. Chapter 3 details the proposed approach, offering an overview of the developed method. In Chapter 4, the focus shifts to incident forecasting using LSTM, presenting the dataset, the selection of analyzed institutions, and the details of the conducted experimental study. In Chapter 5, the results obtained are discussed, including the performance of the LSTM model in relation to the experiments described in Chapter 4, as well as the analysis of anomalies in malware incidents. Finally, Chapter 6 presents the final conclusions, reflecting on the results achieved and highlighting the importance of organizational maturity for the effective adoption of the method. Additionally, future improvements are proposed, such as exploring new neural network architectures and including additional variables to enrich the model and expand its applicability.

2 FUNDAMENTAL BACKGROUND

In this chapter, fundamental topics for understanding this work are addressed, including a detailed analysis of information security incidents, with an emphasis on malware incidents. The role of CSIRTs in responding to these threats is examined, and the concepts of machine learning, time series, ARIMA models, and LSTM are introduced, as these will be used in predicting malware incident rates. Additionally, a Section on related work is presented, highlighting the main studies in the field of security incident prediction and how this work is distinct and relevant compared to others.

2.1 Cyber Security Incidents

The definition of security events varies among the primary CSIRTs in Brazil and globally. The International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) 27000:2018 standard, developed by the ISO and the IEC, provides the definition of a security incident as "an unwanted or unexpected event or series of events related to information security that have a high probability of compromising business operations and endangering information security" [15].

The Security Incident Response Center of the Brazilian National Research and Education Network (CAIS-RNP) defines a security incident as any event, whether confirmed or suspected, that compromises the integrity, confidentiality, or accessibility of RNP's information [16]. This includes scenarios where there are suspicions or tangible proof of unlawful disclosure of data. The Brazilian National Computer Emergency Response Team (CERT.br) categorizes any negative occurrence pertaining to the security of computer systems or networks as a security incident [17]. Furthermore, according to the European Union Agency for Cybersecurity (ENISA), a security incident is defined as an occurrence that impacts the confidentiality, availability, integrity, and authenticity of networks and services [18]. Although definitions may vary among groups, there are several shared characteristics that everyone agrees upon, as evidenced. Given that the incidents used in the experiments in this work were acquired from RNP, we will employ the CAIS-RNP criterion.

Once the definition of an incident for the organization has been established, it is necessary to create a taxonomy that enables the categorization of events based on their kind, subtype, severity, and the necessary data to be recorded for each occurrence. This includes determining which parts are required and which are optional. Several sources can provide guidance for developing a personalized classification system. The taxonomy used by CAIS-RNP was derived from proposals put forth by ENISA, the Forum of Inci-

dent Response and Security Teams (FIRST) [19], and the Computer Incident Response Center Luxembourg (CIRCL) [20]. It was further refined with information provided by [21]. A well-organized taxonomy enables a precise comprehension of the characteristics and seriousness of security incidents, facilitating a prompt and effective reaction. Furthermore, the thorough classification of occurrences enables the recognition of patterns and trends, hence enhancing the organization's security strategy. When evaluating events, it is important to examine factors such as the impact on business operations, the scope of data compromised, and the level of vulnerability exploitation. Regularly reviewing and updating the classification allow to accurately represent changes in the threat landscape and the organization's priorities.

When managing security incidents, multiple standards provide guidance on the essential information to document and later share regarding a security issue. The often employed standards are Incident Object Description Exchange Format (IODEF), Structured Threat Information eXpression (STIX), and Extended Abuse Reporting Format (X-ARF) [22]. IODEF offers a standardized structure for summarizing security incidents, hence promoting the exchange of information across diverse organizations. STIX is utilized to depict cyber threat information in an organized way, facilitating the examination and dissemination of data on attacks, vulnerabilities, and other pertinent security information. X-ARF is a standardized format used for reporting incidents of Internet abuses, including spam, phishing, malware, and other unwanted actions.

The incidents examined in this research adhere to the IODEFv2 standard to disseminate information about incidents to their clients. The Internet Engineering Task Force (IETF) developed this standard, which is used by various CSIRTs globally [23]. The IODEFv2 categorizes incident fields into two groups: mandatory and optional. The mandatory fields are those that must be completed for the incident record to be considered genuine according to the standard. The optional fields are those that can be completed depending on the presence of information and the need for specificity in the occurrence. This demonstrates the versatility of the standard and its ability to accommodate various security incident scenarios. Figure 1 illustrates the architecture of the IODEFv2 standard, highlighting only the mandatory components, as described by Colome, Nunes, and Silva in their work on case-based cybersecurity incident resolution [24].

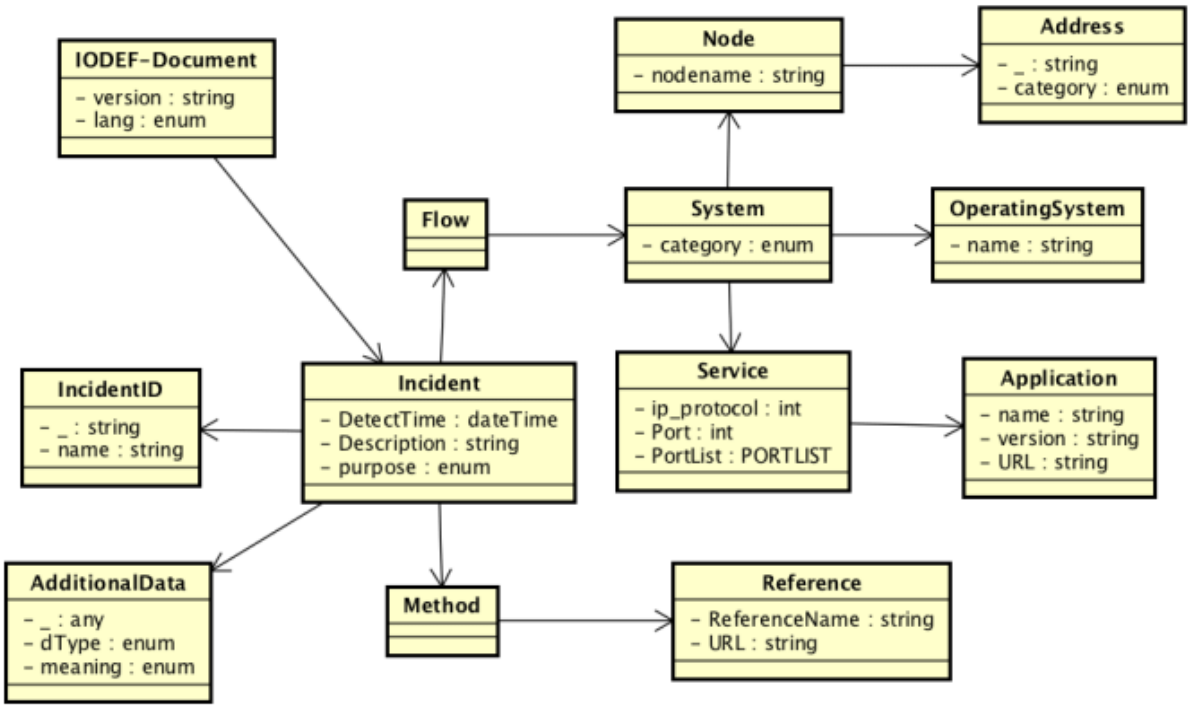

Figure 1 – The IODEFv2 representation. Source: [24]

According to Tufte [25], documenting these incidents allows organizations to carry out various internal and external actions, such as:

- Enhanced Incident Response: recording incident data enables organizations to better understand the cyber threats they face and develop more effective strategies to tackle them.
- Compliance with Legal Obligations: the Budapest Convention, for example, requires country members to implement measures to combat cybercrime, and as Brazil is a member, it must implement these measures.
- Improvement of Security Policies: the analysis of recorded incidents helps to identify vulnerabilities and flaws in cybersecurity policies and practices, enabling enhancements.
- Continuous Learning: the recording and analysis of incidents allow learning from past experiences, contributing to the continuous evolution of cybersecurity strategies.

All the points mentioned above are components of the security incident response process. The progression of a security incident, as outlined by the National Institute of Standards and Technology (NIST) [1], consists of four primary stages. The Preparation

phase is responsible for establishing reaction capabilities by developing policies, detecting tools, and awareness. The Detection and Investigation phase involves the identification and evaluation of incidents through the utilization of monitoring and forensic investigation techniques. Afterwards, the Eradication and Recovery phase eliminates dangers and reinstates systems while incorporating precautionary measures. In conclusion, the Post-Incident phase evaluates the incident response in an effort to enhance it in the future. This includes updating procedures and conducting a review of lessons learned. Meanwhile, for the SANS Institute, the cybersecurity incident lifecycle comprises six separate phases: Preparation, Identification, Containment, Eradication, Recovery, and Lessons Learned. SANS highlights Identification as a distinct stage focused on identifying the incident, differentiating it from the later stages of analysis and investigation. In addition, Containment is regarded as a distinct phase, emphasizing the crucial significance of isolating and reducing the impact of the event on the environment [26].

In Brazil, there is no dedicated regulation for the documentation of security incidents. However, several governmental decrees tackle matters concerning information security, hence emphasizing the significance of such measures.The National Information Security Policy, established by the Federal Government Decree No. 9637 of 2018 [27], provides principles for governing information security in the federal public administration. The National Cybersecurity Strategy (E-Ciber) has been approved by the Federal Government Decree No. 10222 of 2020 [28]. This decree provides rules for ensuring national cybersecurity. Furthermore, the General Data Protection Law (LGPD) - Law No. 13.709/2018 emphasizes the significance of documenting and handling security breaches, particularly those concerning the safeguarding of personal information [29].

2.1.1 IT Governance and Management

Frameworks such as ITIL and COBIT have established key guidelines for best practices in IT governance and management. They have been widely adopted to ensure that IT services are aligned with business objectives, while also promoting operational efficiency and risk control [13] and [14].

ITIL provides detailed rules for IT service management to align IT services with business needs. It emphasizes the importance of successful incident management to minimize the effects of incidents and quickly restore services, ensuring operation continuity [14]. The use of a well-defined taxonomy and standards, such as IODEFv2, to classify and communicate incidents is in line with ITIL practices, which value process documentation and standardization for continuous improvement. Additionally, ITIL defines specific activities for information security, including: policy management (creation, review, and implementation of security policies); information classification (analysis and classification of data); security controls (implementation and improvement of controls and risk re-

sponses); security incident management (detection, response, and resolution of incidents, with the establishment of response teams and escalation procedures); security reviews (audits and penetration tests); security monitoring (management of breaches and major incidents); and security reporting (analysis and actions to reduce the volume and impact of incidents). These activities emphasize the importance of standardized and structured processes to ensure the protection of IT services.

COBIT is a framework for IT governance and management that helps companies create value from IT while maintaining a balance between realizing benefits and optimizing risk levels and resource use. To ensure that IT risks are managed effectively, the practices supporting COBIT principles include detailed categorization of incidents and the use of recognized standards for incident documentation and information sharing [13]. Additionally, the principle of continuous monitoring and evaluation promoted by COBIT is reflected in the regular review and updating of the incident taxonomy. This ensures that information security practices are always adapted to changes in the threat landscape.

2.1.2 Malware

The malware concept can be referenced by several names. Examples include malicious software, malicious code, and malcode. Various definitions have been offered to describe malware. For example, in the work by Christodorescu and Jha [30], they describe a malware instance as a program with malicious purpose. McGraw and Morrisett [31] define malicious code as "any code added, changed, or removed from a software system with the intention of causing harm or subverting the intended function of the system". In our work, we adopt the description of malware provided in [32] and [33], which describes malware broadly as a range of online threats, including viruses, spyware, adware, trojans, ransomware, and other types of harmful software.

According to [34], the methods for potential malware infection encompass a long list of possibilities. The main ones are the two most common ways malware accesses a system, the Internet, and email. In other words, whenever one is connected online, it is possible to be infected. A user can get their computer infected by malware by browsing compromised websites, downloading game demos, infected music files, installing unknown plugins, configuring software from risky sources, opening malicious email attachments, or generally downloading any content from the Internet to the device without the protection of antimalware security software.

Malware can be classified into various categories as per [35] and [36].

- Adware: software that is supported by advertising and is programmed to automatically play, show, or download adverts on a computer. This commonly happens following the installation of malicious software or when utilizing a certain program.

This type of malicious malware is commonly incorporated into free software apps. Typical adware programs encompass free games and peer-to-peer clients.

- Spyware: software installed without the user's knowledge. The main objective of this type of software is to transmit the user's confidential personal information to an attacker
- Virus: malicious code created with the aim of changing the default behavior of a program within the operating system. To work, the virus depends on human action.
- Worms: a type of malicious software that autonomously propagates over networks, replicating itself from one system to another. Unlike viruses, worms possess the capability to self-replicate without necessitating any user intervention. They commonly use vulnerabilities in security systems to infect and cause damage, such as by deleting or encrypting files.
- Trojan: malicious software capable of disguising itself as common programs to infect a device. Once the device is infected, non-authorized actions are executed. This type of malware is typically used to create a backdoor and allow the malicious user to control the device.
- Ransomware: type of malware that locks the data or devices of a victim, holding them hostage. The attacker threatens to keep them locked until the victim pays a ransom.
- Rootkit: software used by cyber criminals to gain control over a target computer or network. Usually, it is also designed to stay hidden from the user, software on the device, and the operating system itself.
- Keylogger: developed for keystroke logging practice, recording everything the user types on a computer or cell phone keyboard. They are used to silently monitor the computer or cell phone activity while the user uses their devices normally. Keyloggers are used for legitimate purposes, such as feedback for software development, but they can also be used by criminals to steal the user's data.

After a device is infected with some type of malware, several consequences can occur [37]. Data theft is one of the most serious consequence, where personal, banking, passwords and other confidential data can be stolen and used for fraudulent purposes. Additionally, data loss is a possible consequence, with important files becoming corrupted, lost or encrypted. In the Brazilian context, several universities, both public and private, have already been targets of ransomware [38] and [39], highlighting the importance of preventive measures and effective response.

2.2 CSIRT

The term CSIRT, or Computer Security Incident Response Team, was established in the 1990s by Request for Comments (RFC) 2350 [40]. A CSIRT is also known as a Computer Incident Response Team (CIRT), Computer Emergency Response Team (CERT), Security Incident Response Team (SIRT), and Network Incident Response Team (ETIR), among others. National teams may be referred to as National Cyber Security Centers (NCSCs), which by law often have the CSIRT role and provide additional services for the country. In Brazil, the national CSIRT is the Government Cyber Incident Prevention, Treatment, and Response Center (CTIR.GOV) [41].

According to RFC 2350 [40], a CSIRT serves as an organization at the forefront of combating evolving cyber threats. This team is responsible for managing cybersecurity incidents by monitoring security events, analyzing them, and taking actions. These events include data breaches, malware attacks, and other malicious activities impacting cybersecurity. Additionally, it focuses on preventing incidents by developing strategies to mitigate occurrences, implementing security protocols, providing training, and collaborating with entities to enhance security measures. The main goal of a CSIRT is to minimize the impact of incidents and swiftly restore operations.

A CSIRT can be found at various levels, from national and governmental to private companies and academic institutions. Each level has distinct responsibilities and focuses, but all aim at the same goal: protecting their constituency [40]. RFC 2350 outlines a series of best practices that a CSIRT should follow, such as the CSIRT's name, contact methods, operating hours, address, mission, vision, values, target audience, authority, types of services offered, and types of incidents they handle, as well as how these incidents should be reported. The document also contains information on the technical composition required to create a CSIRT, whether it is at the initial, intermediate, or advanced level.

Currently, FIRST, in collaboration with ENISA, hosts and continuously improves a CSIRT services framework [42], [43], which is a high-level document that describes the activities carried out by CSIRTs. These activities are organized into five main service areas, which are further divided into services, functions, and sub-functions. A CSIRT can choose which services and functions are relevant and organize them into its own service structure.

Typically, a minimal set of services for CSIRTs usually includes those highlighted in bold in Figure 2, as outlined in the FIRST service framework [43].

Figure 2 – FIRST Services Framework. Source: [43]

According to information from FIRST [42], there are no exact numbers available on the amount of technical staff needed to form a CSIRT team, as each CSIRT team is different, works in a different environment, and has different sizes. However, based on the collective experience of the CSIRT community, ENISA has provided a document [43] where the following values have proven to be a good approach:

- To deliver two main services, Incident Response and Vulnerability Coordination: a minimum of 4 full-time team members.
- For a full-service CSIRT that operates only during business hours and maintains its own systems: a minimum of 6 to 8 full-time staff members.
- For a fully equipped 24/7 operation (3 shifts per day), the minimum is 12 full-time staff members.

These numbers include redundancies for illness and vacations.

The National CSIRT of Thailand, for example, operates with 30 people to serve all clients and services offered to the community [42], while the Luxembourg CSIRT, CIRCL, which serves the country's private sector, has 17 members [44]. In Brazil, Cert.br does not disclose information about the number of staff members, and CAIS-RNP, which is the CSIRT of the Brazilian Academic Network, has currently more than 40 employees working in different areas, with approximately 4 people dedicated exclusively to the services delivered.

FIRST [42] and ENISA [43] recommend that for delivering the two main services of a CSIRT, staff should have technical knowledge in: Internet protocols, Linux, Unix, and Windows operating systems, network infrastructure equipment (router, switches, DNS, proxy, mail, etc.), Internet applications (SMTP, HTTPS, FTP, telnet, SSH, etc.), security threats (DDoS, phishing, defacing, sniffing, etc.), and risk assessment and practical implementations of ISO 27000 security standards. The main tools used by a CSIRT for delivering these services should be related to incident recording, analysis, and response, as well as the identification and coordination of vulnerabilities in their clients.

In this context, the predominant solution adopted by CSIRTs for security incident management is RTIR, as indicated by the University of Kent [45]. Regarding vulnerability coordination, Openvas is the most used tool by CSIRTs, according to [46]. The effectiveness of Openvas was individually evaluated in [47] and, in a subsequent comparison with its open-source competitors, it achieved the best results, as reported in [48]. However, it is important to emphasize that as a CSIRT matures, new services are offered to the community, and therefore, other solutions should be incorporated into its portfolio.

A CSIRT, once established with well-structured services and tools, should also periodically assess its maturity using the framework created by the Open CSIRT Foundation, known as the Security Incident Management Maturity Model (SIM3). This reference model framework evaluates CSIRTs maturity, addressing aspects such as governance, documentation, performance, and measurement [49]. The SIM3 model is composed of 44 parameters distributed across four categories (Organizational, Human, Tools, and Processes) and provides a measurement scale from 0 to 4 for each parameter. This allows for a precise assessment of the team's maturity and the identification of areas needing improvement. SIM3, as described in [50], is used by national CSIRTs, ENISA, FIRST, and other organizations, supporting the development and enhancement of team capabilities and ensuring a clearly defined framework to achieve their goals. FIRST even uses this reference model during its membership process [51], where the candidate for membership must demonstrate that the organization adopts a series of procedures described in SIM3 to be approved in the process.

2.3 Machine Learning

Artificial Intelligence (AI) includes the field of Machine Learning (ML), which is recognized for its ability to adjust and learn from the information it receives. A standard machine learning process involves three stages: feeding data to make predictions or categorizations, evaluating the results, and enhancing the algorithm by incorporating feedback to improve its effectiveness. Machine learning models are categorized into unsupervised, supervised and semi-supervised learning approaches. In supervised machine learning, algorithms rely on labeled datasets for training purposes. This type of learning algorithm

is most common in classification and regression problems. These datasets consist of a set of input variables X and a set of labels or target values Y. The main goal is to find a function or procedure $f(.)$ such that $f(X) = Y$. This type of machine learning is the simplest to use, but it has the disadvantage of requiring that each element of X have a corresponding element in Y, which is often unrealistic [52].

To overcome the need for labels in supervised machine learning, an algorithm can instead be trained to identify or classify patterns found in data without using labels. This practice is called unsupervised machine learning [53] and requires a less restricted dataset at the expense of a more complex training and evaluation process. Some unsupervised problems include clustering and dimensionality reduction. Semi-supervised machine learning serves as a middle ground between supervised and unsupervised learning. There are instances where data lacks labeling, making it challenging to use supervised learning effectively. In such cases, a machine learning model can utilize a portion of labeled data to grasp patterns and then apply them to the entire dataset, enabling predictions for unlabeled data [52].

2.3.1 Time Series

According to [54], a time series consists of a sequence of data collected over time. The characteristics of time series data include substantial volumes of data, high dimensional complexity, and the need for frequent updates. Moreover, this data, due to its numerical and continuous nature, is generally analyzed as a complete unit, rather than as isolated numerical data.

Time series modeling has always played a central role in academic research, being crucial for applications in areas such as climate modeling, biology, and medicine, as well as in decision-making in the retail sector and finance, among other examples, as cited by [55]. Thus, a time series can reveal a rich variety of information, such as trends, seasonality, and stationarity. The trend in a series is identified by a constant increase or decrease in values over time [56], while seasonality refers to the recurrence of patterns at regular intervals [57]. A time series is classified as stationary when its mean and variance remain stable over time, indicating the absence of trends and seasonality [58]. Stationarity is a crucial property of time series because dealing with non-stationary series involves understanding several variables associated with their trends and seasonality. On the other hand, a stationary series exhibits more predictable behavior, regardless of the specific moment of observation. For this reason, many analytical tools for time series assume that the series is stationary, requiring special procedures otherwise [58]. The stationarity of a series can be determined by visually observing its behavior, and various tools can be used to detect the stationarity of a series, such as stationarity tests and unit root tests.

Methods such as ARIMA and LSTM neural networks are widely used to model

and predict temporal behaviors [59]. Both methods offer distinct approaches to handle the complexities of time series, adapting to their specific characteristics such as trends and seasonal patterns.

The ARIMA model is a statistical model used for analysis and forecasting of time series data. Developed by George Box and Gwilym Jenkins in 1970 [60], this methodology combines three components: autoregression (AR), differencing (I), and moving average (MA). The AR part captures the relationship between an observation and a specified number of lagged observations. The differencing step helps achieve stationarity by subtracting previous observations from current ones. The MA component models the dependency between an observation and the residual errors from a moving average model applied to lagged observations. By integrating these elements, ARIMA can model various time series data, capturing trends, seasonality, and cyclic patterns.

LSTMs, in contrast, are recurrent neural networks (RNN) created to retain information over extended periods. Unlike traditional RNNs that can face challenges with long term connections due to the vanishing gradient dilemma, LSTMs employ a gating system to manage information flow [61]. This system comprises three gates; the input gate, for determining which values to update; the forget gate, responsible for deciding which values to discard; and the output gate determining the cells output. By allowing information through these gates, LSTMs can effectively preserve details over time making them well suited for tasks like language modeling, speech recognition and predicting time series data. Conceived by Sepp Hochreiter and Jürgen Schmidhuber in 1997, LSTMs have emerged as a component in learning because of their efficacy in dealing with sequential data [62].

2.4 Related Work

In recent years, there has been an increase in the number of studies on the prediction of security events and incidents. These approaches aim to provide security teams with a strategic advantage, enabling them to take proactive measures and enhance the efficiency of their operations. To achieve this goal, the studies utilize various methodologies and explore different types of neural networks and statistical models.

The study in [63] proposed a framework based on bidirectional recurrent neural networks with LSTM (BRNN-LSTM) to predict the amount of malicious IP flows. This study deviates from the usual statistical approach and develops a deep learning framework utilizing bidirectional recurrent neural networks with long short-term memory (BRNN-LSTM). The empirical study demonstrated that BRNN-LSTM achieves significantly higher prediction accuracy compared to statistical approaches. Experiments were conducted using data collected from a honeypot. The research by Husák et al. [64] ad-

dresses the use of prediction and projection methods in cybersecurity, highlighting four main tasks: attack projection and intent recognition, intrusion prediction, and network security situation forecasting. The study evaluates discrete models, such as attack graphs, Bayesian networks, and Markov models, as well as continuous models, including time series and grey models. The paper highlights the emerging application of machine learning and data mining, which are promising in a constantly changing environment such as cybersecurity. In their study, ARIMA and exponential smoothing are used to predict the total number of alerts in a network. Both studies in [63] and [64] use data sources that contain false positives, which can impair the estimates' quality.

The study [65] advances in the prediction of cybersecurity events, introducing the Tiresias system, which applies RNNs to anticipate future actions in a system based on past observations. Tested with a massive dataset, Tiresias demonstrated high accuracy in predicting the next event in a system, and its long-term memory was essential, outperforming simpler methods. Additionally, the system includes mechanisms for maintaining the model's stability over time and for reevaluation and readjustment in case of drops in prediction accuracy. The experiments were conducted using data collected from a commercial Intrusion Prevention System (IPS), comprising approximately 3 billion security events. By employing data from a commercial IPS, it is acknowledged that there may be a considerable presence of false positives, as evidenced in [66].

The work [67] explores a distinct angle in predicting cyber attacks, focusing on incidents verified by analysts. Unlike other research that relies on automated systems, this study utilizes high-quality data from an operational cybersecurity service provider, which reduces the presence of false positives. While the LSTM model is common in time-based predictions, the study opted for Bayesian state-space models to accurately predict events, except during periods of anomalous activity. The advanced forecasting approach provides a basis for resource allocation and effective defense strategies.

The study in [68] presents an integrated system that uses machine learning solutions to attempt to predict cyber attacks before they occur. It employed the supervised machine learning algorithm K-Nearest Neighbors (KNN). The work analyzed data from 16 months of cyber incidents, achieving some accuracy in predicting malware, malicious destinations, and email attacks, demonstrating some robustness of the model.

Mahmood and Abbasi [69] propose a framework based on deep generative models for time series forecasting in noisy contexts, such as cybersecurity. The proposal was to combine generative models (TimeGAN) and autoencoders to handle noisy data series. The framework is extensible and demonstrated increased predictive power for various standard time series models, outperforming generative data augmentation approaches, indicating that the use of generative models improved forecasting in the context of cybersecurity threat intelligence. The works [67], [68], and [69] aim to use incident records generated

under the supervision of analysts, i.e., with a low chance of containing false positives.

Lastly, Zuzcak and Bujok [70] analyse the use of time series to predict SSH attacks, a protocol often targeted by attackers. Using real-world data without manual validation, the research employs methods such as ARIMA, SARIMA, GARCH, and Bootstrapping to predict attacks on honeynets and from specific geographic regions. The paper assesses the feasibility, accuracy, and usefulness of these approaches for different areas of information security.

Compared to the works presented, our proposal stands out based on the following points: unlike most previous research, which uses data lacking detection and verification by security incident analysts, our study involves a large dataset comprising real malware incidents detected and analyzed by CAIS-RNP. This is important because the use of unverified incident or alert data can contaminate the analysis with false positives. Additionally, these incidents were collected from various types and sizes of public universities, constituting a remarkably diverse dataset that enables a more comprehensive and thorough analysis. The incidents are used to create a method that allows CSIRTs or security teams to analyze trends and behavioral variations. The main activity of this method consists of developing a neural network model that performs time series analysis and uses LSTM to make predictions based on security incident history. The proposed method also includes the use of a visualization tool that allows the security analysts to carry out a detailed inspection and interpret the information related to the incidents and their respective forecasts. Once processed and made available, security analysts use this information in an analytical and decision-making approach, thus promoting a proactive attitude and creating an optimized operational method that assists in the prevention and response to security incidents.

3 PROPOSED APPROACH

The proposal consists of a method that uses historical data about malware security incidents to assist CSIRTs and security teams in analyzing trends and behavioral variations. This method's core activity involves developing a neural network model that uses time series analysis with LSTM to generate predictions based on historical incident data. Additionally, it includes a visualization tool that allows security analysts to thoroughly examine and interpret data related to the incidents and their forecasts. The processed information is then used in an analytical and decision-making approach, promoting a proactive stance and enhancing the ability to prevent and respond to security incidents.

This proposal is closely aligned with key ITIL activities related to information security, including Security Incident Management, which involves detecting, responding to, and resolving security incidents, as well as implementing incident management practices and escalation procedures. The use of historical data and predictive models contributes to Security Monitoring, enabling the proactive identification and management of potential breaches and incidents. Moreover, the visualization and analytical tools support Security Reporting by offering structured analyses that aid in decision-making and enhance the effectiveness of incident response. These activities reflect ITIL's emphasis on process standardization, continuous improvement, and alignment of IT services with organizational needs.

Figure 3 illustrates the predictive process from a broad perspective. For a comprehensive analysis, tables that describe each stage of the process, with detailed information, are available in the Appendix and should be consulted for a deeper understanding. These tables were developed based on best practices in Business Process Management (BPM) [71], a widely used methodology for modeling, monitoring, and optimizing organizational processes. BPM offers a structured approach, ensuring that all stages of the process are clearly mapped, highlighting the activities, required inputs, and expected outputs. The use of BPM also allows the malware incident prediction method to be adaptable to different environments and levels of organizational maturity, providing flexibility for adjustments according to the existing infrastructure. The process described in each table follows the BPM logic, in which activities are accompanied by well-defined inputs, descriptions, and outputs, promoting a clear and objective view of the incident cycle, from registration to resolution.

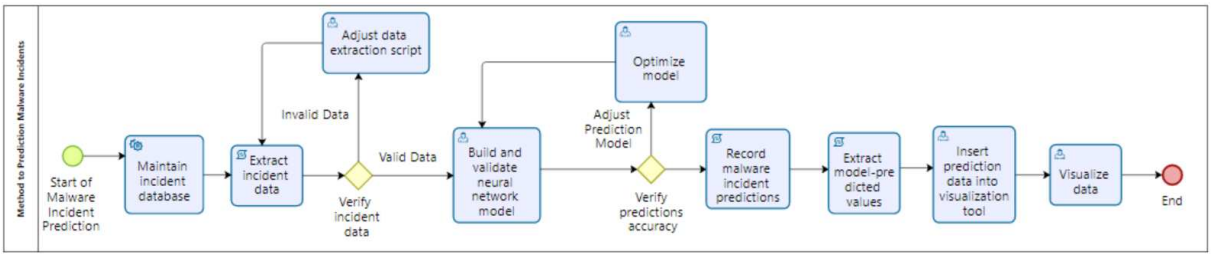

Figure 3 – Macro Process of the Method for Predicting Malware Incidents.

This proposal aims to contribute to the existing literature by creating a model that can be easily adapted to other cybersecurity contexts, such as other CSIRTs, making it a versatile solution for protection against digital threats. Furthermore, the research provides valuable insights into the use of LSTM and ARIMA for malware forecasting in academic environments, which are often targets of cyber attacks due to their open and diverse network infrastructure.

4 FORECASTING INCIDENTS WITH LSTM

This chapter provides a detailed analysis of the experiment conducted to assess the effectiveness of LSTM in predicting security incidents, as LSTM-based forecasting is a central element in the process outlined in Chapter 3. Details of the experiment, including the methodologies employed, the datasets used, and the evaluation metrics applied, are explored in this chapter. All the used data was obtained from RNP, which keeps incident data from public higher education institutions in all Brazilian regions, with the experiment being organized in two scenarios. In Scenario 1, a specific time series is created for each analyzed institution. In Scenario 2, each RNP's PoP encompasses a distinct time series. A PoP covers all institutions located within a particular federated state in Brazil, ensuring complete coverage and enabling a detailed and regionalized analysis of the data.

In our research, the supervised learning paradigm is used for the analyzed time series. According to [72], the main advantages of using supervised learning in an LSTM network are related to aspects such as the ability to model complex sequences, provide accurate predictions, receive clear feedback on predictions, and, lastly, its generalization capability. The use of recurrent neural networks, like LSTM, has proven promising in time series applications across various domains, making this technique a valuable option for cybersecurity data analysis, as presented in [63]. This experiment was designed to address the following questions:

- How accurate is an LSTM-based model in forecasting the number of malware incidents that will occur in the next week, considering occurrences from previous weeks?
- How does the institution's size (and its number of weekly incidents) influence the error rate in prediction?
- Does increasing the number of weeks passed as input to the LSTM model reduce prediction errors?
- Can more accurate predictions be obtained by extending the training period used in the model?
- Is a model based on LSTM more accurate than a ARIMA-based one ?

4.1 Dataset and Institution Selection

The malware incident data analyzed was obtained in collaboration with the custodians of the data, RNP. This partnership facilitated access to detailed information and

enabled a comprehensive analysis of security incidents in various public higher education institutions in Brazil.

Figure 4 provides an overview of the procedures adopted by RNP for identifying, classifying, and analyzing security incidents. The detection of these events is based on opened tickets, which may be either internal, generated by malicious activity sensors, or external, originating from partner CSIRTs. Once an event is detected, the team logs it in a monitoring system and submits it to a screening process to evaluate whether it qualifies as a security incident. If the event does not meet the criteria for a security incident, it is classified as a false positive, which is crucial to optimize the security monitoring process, focus efforts on real threats, and improve the efficiency of client protection.

Confirmed events are then classified as security incidents based on their nature and severity and subsequently prioritized for efficient handling, taking into account the urgency and impact of each case. Each prioritized incident is then assigned to the organization or department responsible for managing the respective network segment, culminating in the formal registration and designation of the incident, preparing it for resolution by the appropriate team within the organization.

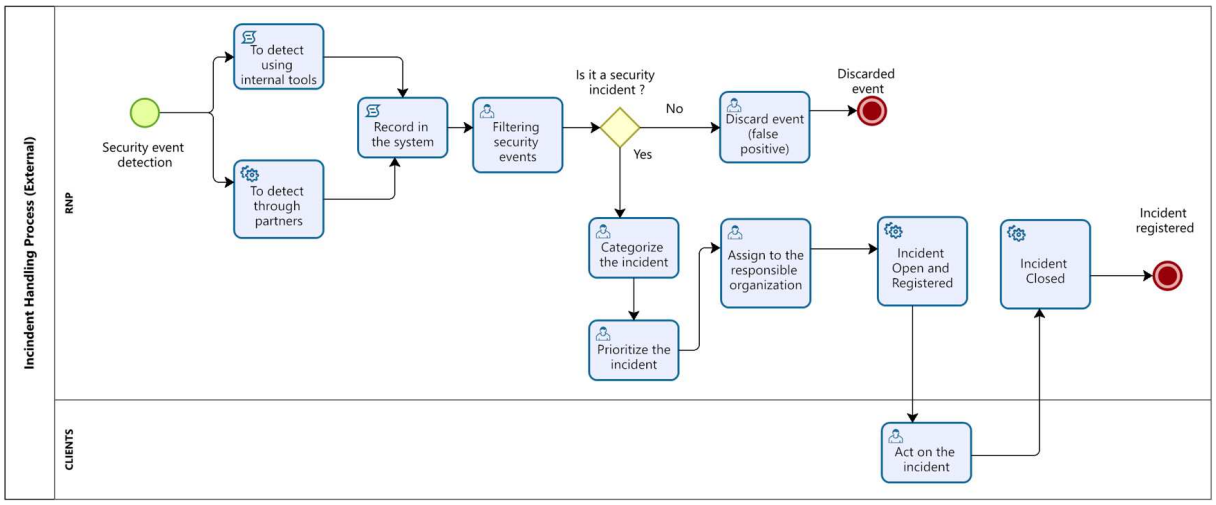

Figure 4 – Macro Incident Handling Process in the Brazilian National Research and Education Network.

For the experiment, the incidents were organized into two distinct scenarios. In scenario 1, data were collected over three years in a production environment, involving approximately 30,000 malware incidents across various types of higher education institutions. This phase of the research did not include all institutions monitored by RNP, as the focus was on assessing the impact of the institution's size on incident prediction. Given the large number of institutions in the database, over 1500, an initial analysis was conducted on a subset of 10 institutions, divided into two equally sized groups to facilitate data management and analysis. The first group includes the five institutions with

the highest number of public IP addresses and weekly malware incidents over three years, representing the larger institutions. The second group consists of the five institutions with the fewest public IP addresses, yet still recording at least one malware incident per week, classified as smaller institutions. Institutions with fewer than one incident per week were considered inadequate for the study due to insufficient data and were thus excluded from this work. Table 1 details the institutions selected for the research.

Table 1 – Overview of incident figures in selected institutions for a 3-year period.

Institution	Average Number of Incidents Per Week	Standard Deviation
larger_01	16.69	9.36
larger_02	70.85	90.24
larger_03	40.67	22.18
larger_04	19.23	18.56
larger_05	16.19	12.15
smaller_01	1.03	1.71
smaller_02	1.01	2.46
smaller_03	1.05	2.08
smaller_04	1.04	1.98
smaller_05	1.01	1.75

In Scenario 2, data were collected over a period of four years in a production environment, involving approximately 100,000 malware incidents at various higher education institutions in Brazil. In this scenario, experiments were conducted with all client institutions of the RNP, with data being collected and categorized by Brazilian federal states, which are represented in the RNP's system as points of presence (PoP). The motivation for transitioning to this new scenario was to gather data from various institutions which, when aggregated, reduce irregularities in the behavior of the series and can lead to better prediction outcomes. Table 2 presents the details about the data used in this research.

As previously noted, the incidents analyzed in the experiment were extracted from the malware incident database maintained by RNP. This database stores detailed information about the incidents, enabling efficient organization and facilitating data retrieval for in-depth analysis. To conduct the analysis, queries were performed on the database to extract the recorded incident data, which was then transformed into time series. A separate time series was generated for each institution or point of presence (PoP), with each data point representing the weekly count of incidents recorded at that specific location.

The incident data used in this research is not publicly available from RNP. However, upon request for academic and non-commercial purposes, anonymized data can be provided to professors and students to reproduce the achieved results or conduct new experiments. As the data provided are anonymized, there are no legal implications, ensuring the privacy and protection of information. This approach allows the analysis to be reproducible while maintaining a commitment to data confidentiality and security.

Table 2 – Overview of incident figures in each PoP for a 4-year period.

Institution	Mean Number of Incidents Per Week	Standard Deviation
PoP_x01	7.88	6.49
PoP_x02	0.95	2.06
PoP_x03	27.03	19.94
PoP_x04	0.46	1.38
PoP_x05	17.81	12.51
PoP_x06	62.29	48.92
PoP_x07	6.87	9.42
PoP_x08	5.90	8.12
PoP_x09	28.40	17.90
PoP_x10	14.18	14.24
PoP_x11	28.58	36.89
PoP_x12	2.68	2.71
PoP_x13	6.64	6.60
PoP_x14	10.14	7.59
PoP_x15	13.85	15.82
PoP_x16	48.31	37.29
PoP_x17	4.45	3.78
PoP_x18	72.93	88.54
PoP_x19	60.13	36.49
PoP_x20	2.04	3.05
PoP_x21	2.77	4.14
PoP_x22	2.84	3.30
PoP_x23	13.43	11.11
PoP_x24	5.79	7.36
PoP_x25	1.34	1.91
PoP_x26	18.60	13.82
PoP_x27	5.94	7.03

4.2 Experimental Study

To contextualize the scenarios presented in the study, it is essential to clarify how time series were used for modeling and prediction. The time series, composed of sequentially organized data, was structured so that a specific number of weeks was used as input for the LSTM model, which then predicted the value corresponding to the following week. Initially, a period of 4 weeks was used as input to predict the fifth week. Gradually, this period was expanded to 5 weeks to predict the sixth week, until the final test, where 8 weeks were used to predict the ninth week. This approach allowed the model to capture temporal patterns and generate more accurate predictions.

The training of the LSTM model was conducted using a sliding window approach. This process started with a history of 6 months of data, which was gradually increased to longer periods: 1 year, 2 years, and finally 3 years. In each training window, the model

analyzed the historical data to predict the next element in the time series. This methodology was crucial in identifying which time interval provided the best performance in the tests.

To better illustrate this methodology, consider an initial time series composed of the elements x_1, x_2, x_3, x_4. During training, the model was initially fed these four values as input (x_1, x_2, x_3, x_4) and attempted to predict the next element in the sequence, x_5. Since this is a training process, the LSTM algorithm accesses the actual value corresponding to the predicted element (x_5), calculates the error, adjusts its weights, and moves on to the next data window. This configuration was adjusted over time by expanding the historical data and analyzing the results. Figure 5 represents the initial test scenario with the sliding-window approach. The values x_1, x_2, x_3, x_4 constituted the historical data used as input for the LSTM model, which, in turn, predicted the next value in the sequence, x_5. This process was repeated in different training periods, starting with 6 months of history and gradually increasing to longer intervals (1 year, 2 years, and 3 years) to identify the most effective period.

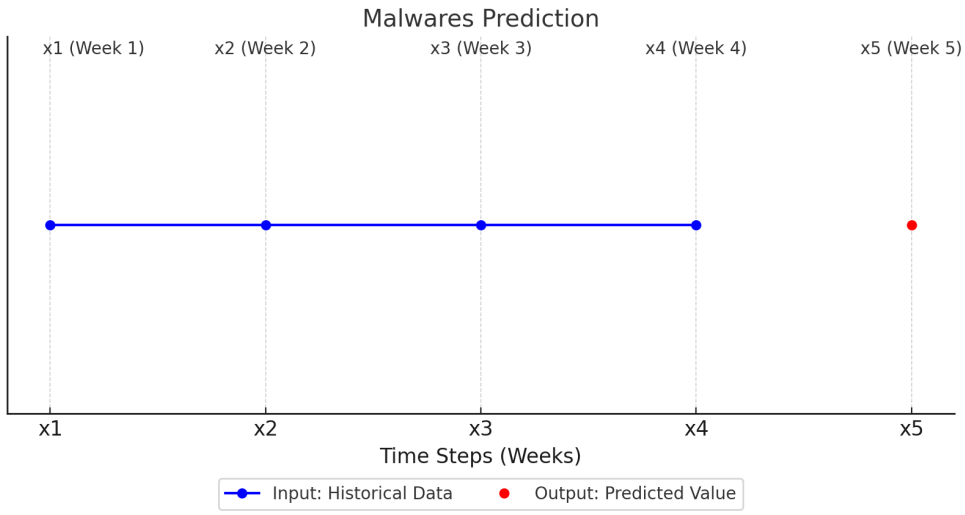

Figure 5 – Malware Incident Predictions.

Based on this foundation, the scenarios in which the study was conducted differ significantly in terms of the number of institutions analyzed, the volume of security incidents considered and the testing period provided. Table 3 presents the training and testing periods selected for Scenario 1, which includes 10 institutions classified as large or small. The training and testing sets for Scenario 2, which focus on incident series for PoPs, are presented in Table 4. These specific training and testing phases for each scenario enable a clear comparison of the effectiveness of different strategies and methodologies employed in each study environment.

Table 3 – Training and test sets to Scenario 1.

Training Period	Test Period
April 2021 to October 2021	November to 2021 to November 2022
October 2020 to October 2021	November to 2021 to November 2022
October 2019 to October 2021	November to 2021 to November 2022

Table 4 – Training and test sets to Scenario 2.

Training Period	Test Period
April 2022 to October 2022	November to 2022 to November 2023
October 2021 to October 2022	November to 2022 to November 2023
October 2020 to October 2022	November to 2022 to November 2023
October 2019 to October 2022	November to 2022 to November 2023

LSTM models were implemented for both scenarios, using an architecture that includes the Adaptive Moment Estimation (Adam) optimization function and Mean Squared Error (MSE) as the loss function. The chosen neural architecture features the Rectified Linear Unit (ReLU) in a hidden layer. The training epochs ranged from 100 to 500, increasing by 100 each time, while the units ranged from 10 to 60, increasing by 10 each time. These variations in hyperparameters and network configurations were designed to explore combinations that optimize prediction accuracy, as discussed in the study by [73]. This approach ensures a systematic evaluation of the LSTM model's performance across different configurations, aiming to identify the most effective architecture and parameter set for the problem domain in question. Specific details of the hyperparameters for the LSTM are presented in Table 5.

Table 5 – Specific hyper-parameters of LSTM to Scenario 1 and Scenario 2.

Parameter	Experimental Choice
Units	10, 20, 30, 40, 50, 60
Units Increments	10
Epochs	100, 200, 300, 400, 500
Epochs Increments	100
Dropout	0

ARIMA models were developed for Scenarios 1 and 2 using the auto-arima function from the Pmdarima library, which automatically adjusts the model parameters [74]. This choice was made due to the complexity and time required for manual parameter configuration. The auto-arima technique simplifies this process by performing multiple adjustments automatically, seeking the best parameters for each dataset. Alongside LSTM, which is the focus of this research, ARIMA was selected as a benchmark for comparison, as it is commonly used in time series analysis as a standard statistical method [75].

The initial experiment aimed to evaluate the best MAE/mean results using only the LSTM technique in Scenarios 1 and 2. Subsequently, for Scenario 1, the best MAE and MAE/mean results were investigated, considering different numbers of weeks as inputs and different sizes of training sets. Next, it was evaluated how well the LSTM and ARIMA techniques could predict outcomes in Scenario 1. Similarly, an evaluation was conducted for Scenario 2, assessing the best MAE and MAE/mean results with different input configurations and, lastly, the performance of the LSTM and ARIMA techniques in that scenario was analyzed. A sequential evaluation was employed to observe how performance evolved with an increase in the training sample size and the number of weeks used as input, designed to emulate a scenario where the model undergoes training for specific durations, such as 6 months, one year, two years, and three years. The objective was to determine if different training periods would improve the model's performance. This method allowed a systematic comparison of the model's efficacy over different time intervals. MAE and the MAE/mean ratio were used to evaluate how well the models predicted the outcomes in both scenarios. To determine the MAE, equation (4.1) was used, where n denotes the total number of observations, y_i represents the values within the time series, $\hat{y}_i$ represents the predictions of the models for each data point, i is an index that iterates through each data point in the sample from 1 to n.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{4.1}$$

The MAE/mean ratio was applied to enable a comparison of the accuracy achieved across different institutions. It provides a relative measure of the average error in comparison to the average magnitude of the observed values. To obtain the ratio, the MAE is divided by the average calculated using the formula $(\sum_{i=1}^{n} y_i)/n$, where we have a time series with n elements and y_i represents the i-th actual element of the time series.

5 RESULTS

This chapter begins by presenting the results for the two experimental scenarios regarding the prediction of malware incidents using LSTM and ARIMA. The first section focuses on comparing the two scenarios, highlighting the forecast accuracy metrics and the influence of different configurations, such as the number of input weeks and the training period duration. The second section then analyzes the implications of these results, offering insights into the applicability and effectiveness of these forecasting techniques across different institutional settings and scales.

In the second section, the implementation of a mechanism for detecting anomalies in time series will be detailed. This mechanism identifies significant deviations between predictions and actual values, serving as an alert system for potential anomalies and enabling quick responses to irregular patterns. This section will present three real cases where the actual number of incidents was significantly different from the predicted values. These cases demonstrate how the proposed model could be used to help analysts more easily identify situations that require greater attention.

5.1 Incidents Forecasting with LSTM and ARIMA

This section details the experimental findings related to the two scenarios tested, as described in Chapter 4. Each scenario was structured to test the efficiency of forecasting techniques under various operational conditions and organizational structures. The best values for the MAE/mean metric for both scenarios are displayed in Table 6. The data were organized per number of input weeks, ranging from 4 to 8.

Table 6 – Results of the Top 3 MAE/mean Metrics: Scenario 1 (10 Institutions) vs. Scenario 2 (All PoPs).

Scenario	Institution	4 weeks	5 weeks	6 weeks	7 weeks	8 weeks
1	larger_05	0.185	0.169	0.186	0.173	0.187
1	larger_01	0.272	0.265	0.256	0.250	0.280
1	larger_03	0.278	0.266	0.240	0.274	0.267
2	PoP_x16	0.069	0.079	0.046	0.064	0.076
2	PoP_x25	0.151	0.156	0.100	0.109	0.107
2	PoP_x22	0.187	0.146	0.115	0.148	0.136

The results for Scenario 1 are the focus of Table 7, which highlights the situations where the LSTM model showed the lowest MAE values. Larger institutions are listed in the first five lines of the table, while smaller institutions occupy the subsequent lines. The MAE/mean ratio in larger institutions varied from 0.17 to 0.29, reflecting that the average

error of the forecasts represents 17% to 29% of the average volume of weekly incidents. In smaller institutions, the errors were significantly higher, ranging from 0.39 to 1.60. The lowest MAE/mean value across all institutions is highlighted in bold in the table.

Table 7 – Best results for the MAE and MAE/mean considering all institutions (Scenario 1).

Institution	Hidden Layer Units	Epochs	Training Period	Input Weeks	MAE	MAE/mean
larger_01	50	100	2 years	7	5.37	0.25
larger_02	20	100	2 years	5	35.02	0.26
larger_03	10	300	2 years	6	12.55	0.24
larger_04	10	100	2 years	4	8.11	0.29
larger_05	30	400	2 years	5	4.17	**0.17**
smaller_01	40	100	2 years	4	0.82	0.82
smaller_02	10	400	1 year	7	1.92	0.65
smaller_03	40	200	2 years	4	1.06	0.39
smaller_04	50	400	2 years	7	0.08	1.60
smaller_05	10	100	6 months	5	0.25	0.89

The data on MAE and MAE/mean address the first two questions mentioned in Chapter 4, highlighting the accuracy of the LSTM model's predictions and emphasizing the impact of the institution's size on its forecasting ability. The results for smaller institutions exhibited inferior performance compared to larger ones, due to greater volatility in their weekly incident rates, characterized by a smaller number of incidents occurring more sparsely over time. Table 7 also provides preliminary answers to other questions proposed in Chapter 4, related to the impact of variations in the number of weeks used as input and the designated training period. With the exception of the six-week configuration, the worst result, all other input settings had the same number of occurrences in the rankings of the best results. The four, five, and seven-week entries each contributed three occurrences of the best results. Figure 6 provides a detailed overview of how the number of weeks influences the accuracy of predictions, showing the best MAE/mean for each configuration at each institution, suggesting a lack of clear correlation between the MAE/mean ratio and the number of input weeks.

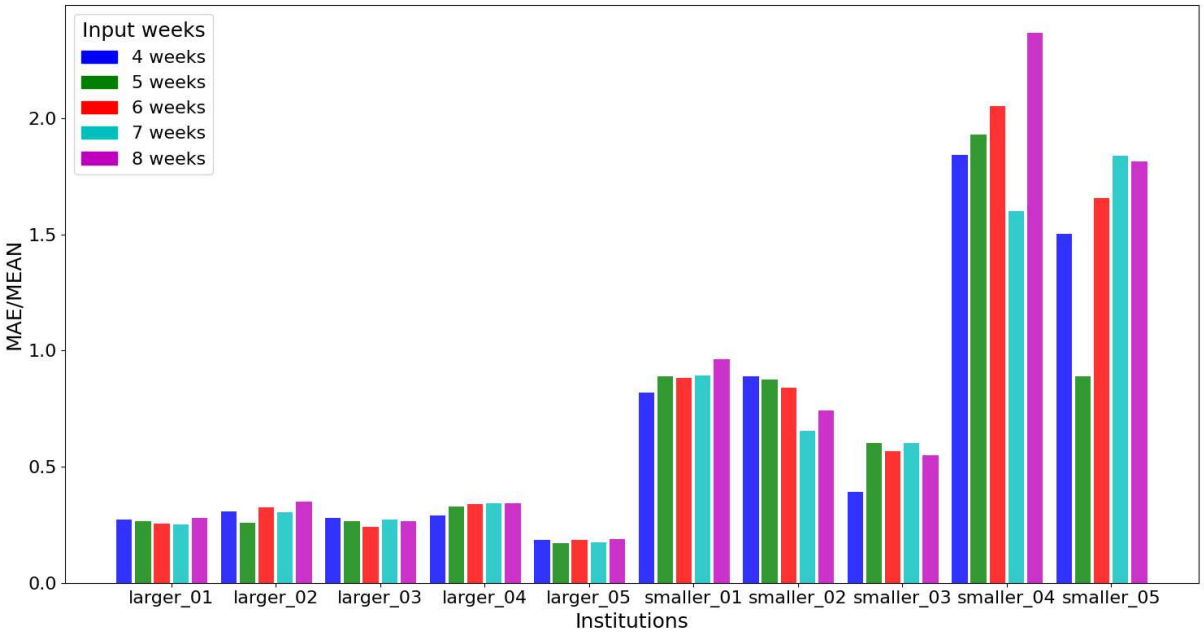

Figure 6 – Best results obtained for MAE/mean considering different numbers of weeks as input (Scenario 1).

As indicated by the results in Table 7, the configuration with a two-year training period produced the best results in eight out of ten institutions analyzed. Although this seems to indicate a clear trend, these results require further investigation. The analysis of Figure 7 reveals important insights about the impact of the length of the training period on the accuracy of forecasts across different institutions. It becomes evident that larger institutions show less variation in the MAE/mean ratio with changes in the length of the training period. This suggests that, possibly due to a larger amount of data or lower volatility in incidents, large institutions are less sensitive to changes in the volume of historical data used for training. Conversely, in smaller institutions, the alteration in the size of the training period has a more significant impact on the MAE/mean ratio. This could indicate that these institutions, due to having less data or greater irregularity in their incidents, rely more heavily on the quantity of training data to generate accurate predictions. Additionally, the configuration with a two-year training period showed slightly better performance in larger institutions, suggesting that a longer training period may be beneficial up to a certain point, but the incremental return might be modest. However, despite these observations, it remains uncertain whether continually increasing the size of the training period will improve the MAE/mean ratio, especially in smaller institutions. This may indicate the existence of a saturation point or diminishing returns regarding the increase of training data volume.

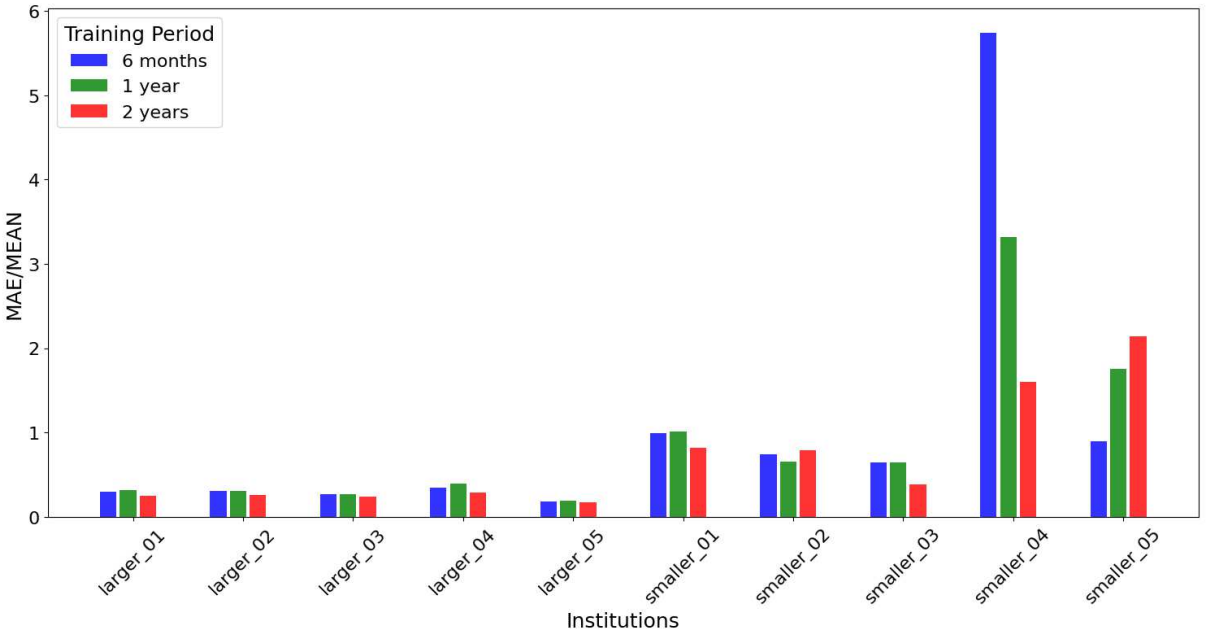

Figure 7 – Best results obtained for MAE/mean considering different training set sizes (Scenario 1).

The subsequent analysis focuses on evaluating and comparing the accuracy of the forecasts made by the LSTM and ARIMA models in Scenario 1, aiming to determine which technique exhibits superior predictive capability. The MAE results achieved by both models are detailed in Table 8. It is observed that the performance of the LSTM model surpassed that of ARIMA by approximately 10% in most institutions. Notably, at the institution smaller_02, the improvement in MAE achieved by the LSTM was 52%. These results demonstrate that LSTM consistently showed higher accuracy in predicting malware incidents compared to ARIMA. However, considering that LSTM is an advanced machine learning technique, the average margin of superiority may be seen as relatively modest.

Table 8 – Best MAE results for evaluating the LSTM and ARIMA models (Scenario 1).

Institution	Best LSTM - MAE	Best ARIMA - MAE
larger_01	5.37	6.11
larger_02	35.02	38.8
larger_03	12.55	14.12
larger_04	8.11	9.14
larger_05	4.17	4.57
smaller_01	0.82	0.92
smaller_02	1.92	2.94
smaller_03	1.06	1.17
smaller_04	0.08	0.09
smaller_05	0.25	0.28

After carry out tests for Scenario 1, we move on to the next step. Predictions were made using data from Scenario 2, which includes all client institutions of RNP. Table 9 presents the data for Scenario 2, including only the best MAE and MAE/mean results for each of the analyzed PoPs. The results for the MAE/mean ratio indicate that, for some PoPs, the predictions were quite accurate, with variations from 0.05 to 0.49 for this metric. As a point of comparison with the data from Scenario 1, variations ranged from 0.17 to 1.60. These values mean that the average error of the predictions for the weekly incident rate in this scenario ranged from 5% to 49% of the average number of incidents. The best MAE/mean value analyzed is highlighted in Table 9 in bold.

Table 9 – Best results obtained for MAE/mean considering different numbers of weeks as input (Scenario 2).

Institution	Hidden Layer Units	Epochs	Training Period	Input Weeks	MAE	MAE/mean
PoP_x01	20	100	6 months	6	1.16	0.15
PoP_x02	60	400	2 years	5	0.21	0.22
PoP_x03	80	400	3 years	6	0.11	0.25
PoP_x04	60	100	3 years	6	3.30	0.12
PoP_x05	80	200	3 years	6	2.14	0.12
PoP_x06	40	300	3 years	6	8.34	0.13
PoP_x07	40	200	2 years	8	1.70	0.25
PoP_x08	20	200	3 years	6	2.47	0.42
PoP_x09	60	200	3 years	6	5.60	0.20
PoP_x10	20	200	3 years	6	2.43	0.16
PoP_x11	60	300	3 years	6	1.43	0.21
PoP_x12	60	400	6 months	8	1.04	0.39
PoP_x13	40	400	2 years	6	3.45	0.14
PoP_x14	40	100	3 years	6	2.32	0.23
PoP_x15	20	100	2 years	6	2.42	0.17
PoP_x16	20	200	3 years	6	3.36	**0.05**
PoP_x17	20	300	3 years	6	4.84	0.10
PoP_x18	20	100	2 years	6	1.22	0.27
PoP_x19	80	200	3 years	6	9.05	0.15
PoP_x20	20	200	1 year	6	0.99	0.49
PoP_x21	40	300	3 years	6	2.21	0.16
PoP_x22	20	100	3 years	6	0.32	0.12
PoP_x23	20	200	3 years	8	1.27	0.45
PoP_x24	80	100	1 year	6	1.98	0.34
PoP_x25	20	300	3 years	6	1.85	0.10
PoP_x26	20	400	3 years	6	0.17	0.13
PoP_x27	60	300	3 years	6	2.10	0.35

In Scenario 2, the impact of varying the number of weeks used as input was analyzed, observing that the best results were obtained with an input of six weeks. Figure

8 provides a detailed view of how this variation influences the accuracy of predictions, highlighting that the six-week input is the most effective for the tested configurations. Furthermore, the figure reveals an apparent correlation between the MAE/mean ratio and the number of weeks used as input, reinforced by the presentation of the top ten best results obtained. This focus on the most successful cases indicates that adjusting the number of weeks of data used as input can be an effective strategy for optimizing the accuracy of predictions.

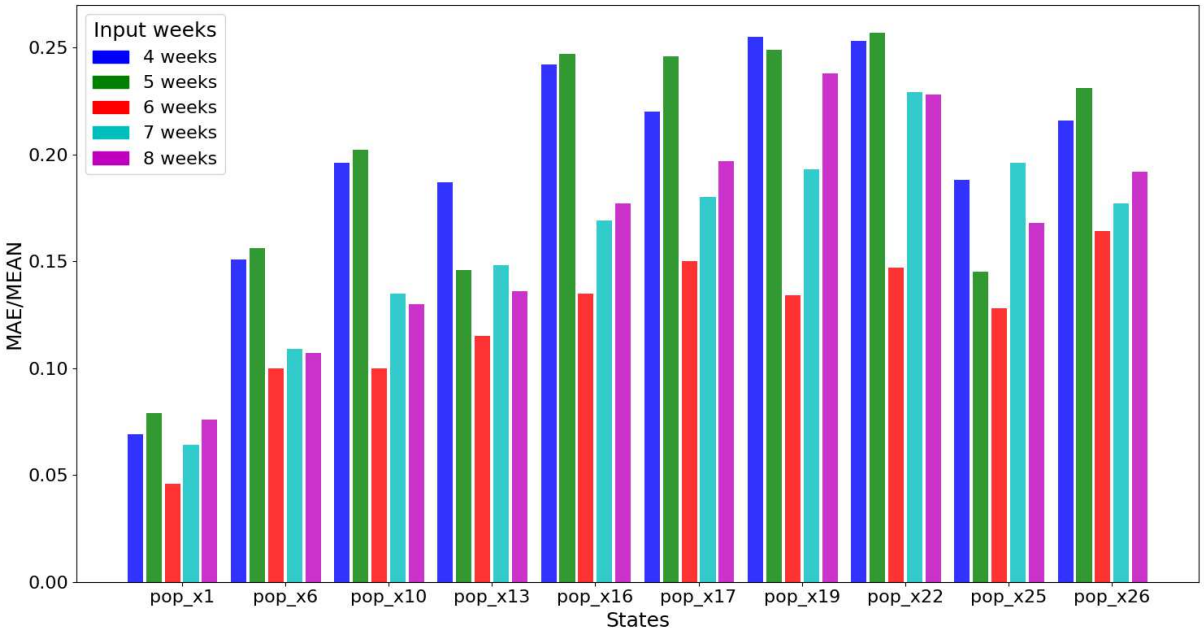

Figure 8 – Best results obtained for MAE/mean considering different numbers of weeks as input (Scenario 2).

The analysis continued in Scenario 2, investigating whether changes in the training period could result in variations in model performance. The data presented in Figure 9 reveal that the three-year configuration for the size of the training set was the most effective, producing the best results in 18 out of 27 PoPs. Although at first glance this may seem to indicate a clear trend, such results require more careful analysis. Figure 9 highlights the best MAE/mean ratio achieved, focusing on the top ten best results obtained, which suggests that a longer training period may be beneficial.

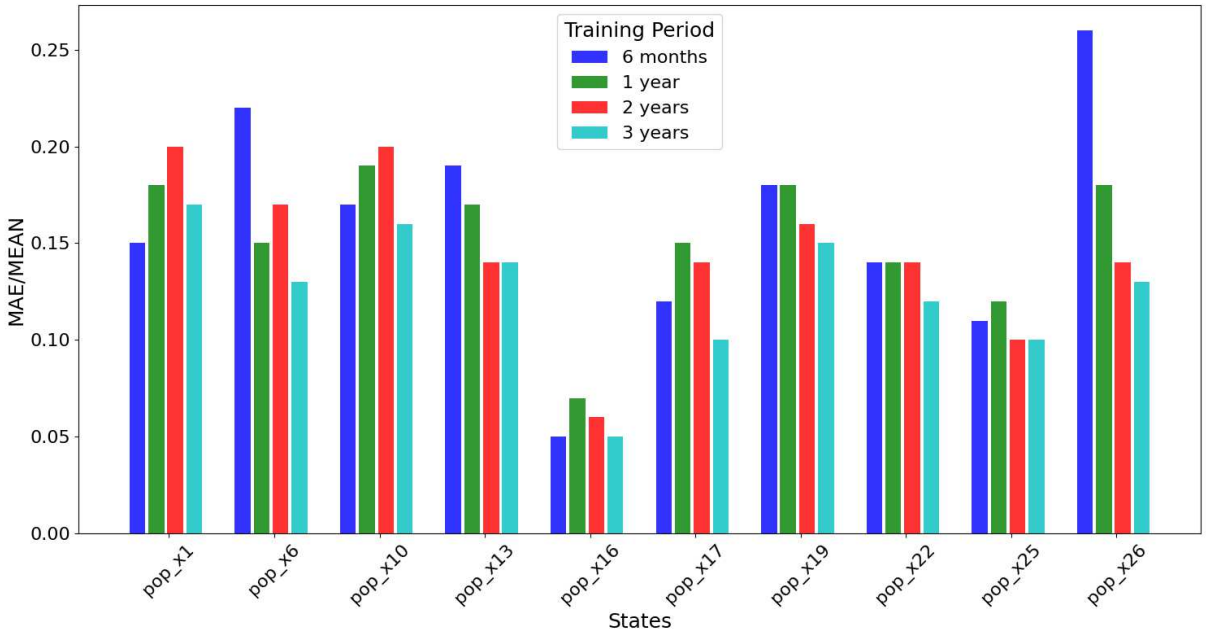

Figure 9 – Best MAE/mean values for the top 10 results obtained considering different training set sizes (Scenario 2).

The analysis of Scenario 2 was further deepened by evaluating the accuracy of the forecasts generated by the LSTM and ARIMA models, with the purpose of determining which technique exhibited superior predictive capability in this expanded context. Table 10 compiles the best MAE results achieved by both models in each PoP. Notably, the MAE results for the LSTM models were approximately 40% lower than those obtained by the ARIMA models in all the analyzed PoPs. This pattern reinforces that LSTM demonstrated consistently higher accuracy than ARIMA in predicting malware incident rates, confirming its superior efficacy in this scenario.

Table 10 – Best MAE results for LSTM and ARIMA models (Scenario 2).

Institution	Best LSTM - MAE	Best ARIMA - MAE
PoP_x01	1.16	2.24
PoP_x02	0.21	0.25
PoP_x03	0.11	0.13
PoP_x04	3.30	7.27
PoP_x05	2.14	5.40
PoP_x06	8.34	15.13
PoP_x07	1.70	2.37
PoP_x08	2.47	3.39
PoP_x09	5.60	10.60
PoP_x10	2.43	3.52
PoP_x11	1.43	2.26
PoP_x12	1.04	1.93
PoP_x13	3.46	5.91
PoP_x14	2.32	5.09
PoP_x15	2.42	3.33
PoP_x16	3.37	6.65
PoP_x17	4.84	9.92
PoP_x18	1.22	1.81
PoP_x19	9.05	16.54
PoP_x20	0.99	1.74
PoP_x21	2.22	4.16
PoP_x22	0.32	0.54
PoP_x23	1.27	2.35
PoP_x24	1.98	3.90
PoP_x25	1.85	3.46
PoP_x26	0.17	0.21
PoP_x27	2.10	3.42

5.2 Analysis of Anomalies in Malware Incident Series

Monitoring fluctuations in malware incident series to detect anomalous patterns can offer security analysts valuable insights, whether by revealing ongoing malware campaigns or demonstrating the effectiveness of previously implemented protective measures. In this context, the data visualization tool, which constitutes the final part of the proposed method, plays an essential role by enabling the quick and easy identification of situations where the actual incident values significantly deviate from the predictions generated by the model. Additionally, the tool can act as an alert mechanism, allowing security analysts to be notified of potential anomalies, thereby facilitating preventive decision-making.

To provide context for the analysis that will be presented next, it is essential to revisit the configuration of the LSTM model that provided the best results and will be employed in this stage. The model was trained using a dataset covering a three-year period and configured to use a 6-week interval as input to predict the following week. The choice

of this approach, based on the 6-week interval, was grounded in previous results, aiming to improve the accuracy of the weekly forecasts. More details about the LSTM model can be found in Chapter 4 and Section 5.1, where its configuration and results were discussed and presented.

The implemented tool enabled an analysis that identified and examined all cases where the difference between the predicted and actual incident rates exceeded 30%, focusing exclusively on situations with five or more incidents. Based on this evaluation, three specific cases were selected to be discussed in this section. The data used covers the period from July 1, 2023, to July 1, 2024, and was chosen based on the PoPs (Points of Presence) with the highest average incident rates over the past four years, as shown in Table 2. The selection of these three cases aimed to exemplify the potential of the implemented tool, demonstrating its significant contribution to anomaly detection in malware incident series within an organization. Currently, the visualization tool is being used in RNP's production environment, enabling continuous evaluation and the implementation of improvements to optimize its effectiveness.

Case 1: The first case refers to a situation observed in PoP_x18. It is important to emphasize that the data presented in Table 11 present a specific snapshot of a broader period, extending from July 1, 2023, to July 1, 2024. This snapshot focuses on the interval between January 21 to March 24, highlighting selected weeks in which significant discrepancies were identified between the model's predictions and the actual values of malware incidents, revealing anomalous situations.

To better understand the data presented in Table 11, its columns are detailed next. The "Institution" column, for example, identifies the PoP that was analyzed. The "Start Week" and "Final Week" columns indicate, respectively, the day on which the first week and the day on which the last week used as input for the model begin, during the six-week interval configured as the forecast input. The "Forecast Week" column, in turn, indicates the day on which the week predicted by the LSTM model begins. The "Real Value" column presents the actual number of malware incidents recorded in the predicted week, while the "Forecasted Value" column shows the estimated value predicted by the model for the analyzed period. Finally, the "Mean" column provides the average of the actual incident values over the six weeks used as input by the model, offering additional context that helps assess the accuracy of the prediction.

Table 11 – Comparison of Predicted and Actual Weekly Malware Incident Values at PoP_x18

Institution	Start Week	Final Week	Forecast Week	Real Value	Forecasted Value	Mean
PoP_x18	Jan 21, 2024	Feb 25, 2024	Mar 3, 2024	2	1.42	0.5
PoP_x18	Jan 28, 2024	Mar 3, 2024	Mar 10, 2024	1	1.79	0.67
PoP_x18	Feb 4, 2024	Mar 10, 2024	Mar 17, 2024	3	2.75	1.17
PoP_x18	Feb 11, 2024	Mar 17, 2024	Mar 24, 2024	12	6	3.17
PoP_x18	Feb 18, 2024	Mar 24, 2024	Mar 31, 2024	12	12.2	5.17

When analyzing the input data from February 4 to March 10, it was found that the LSTM model estimated 2.75 incidents for the week beginning on March 17, while the actual number recorded was 3 incidents. However, for the following period, using input data from February 11 to March 17, the model predicted that 6 incidents would occur in the week beginning on March 24, but the actual number of incidents was 12, representing a significant difference of 100%, as shown in Table 11.

To facilitate the visualization of these discrepancies between the forecasted and actual values, a graphical visualization tool was developed. The tool was implemented using the Python programming language [76] and the Matplotlib library [77]. The graph presented in Figure 10 displays the following information:

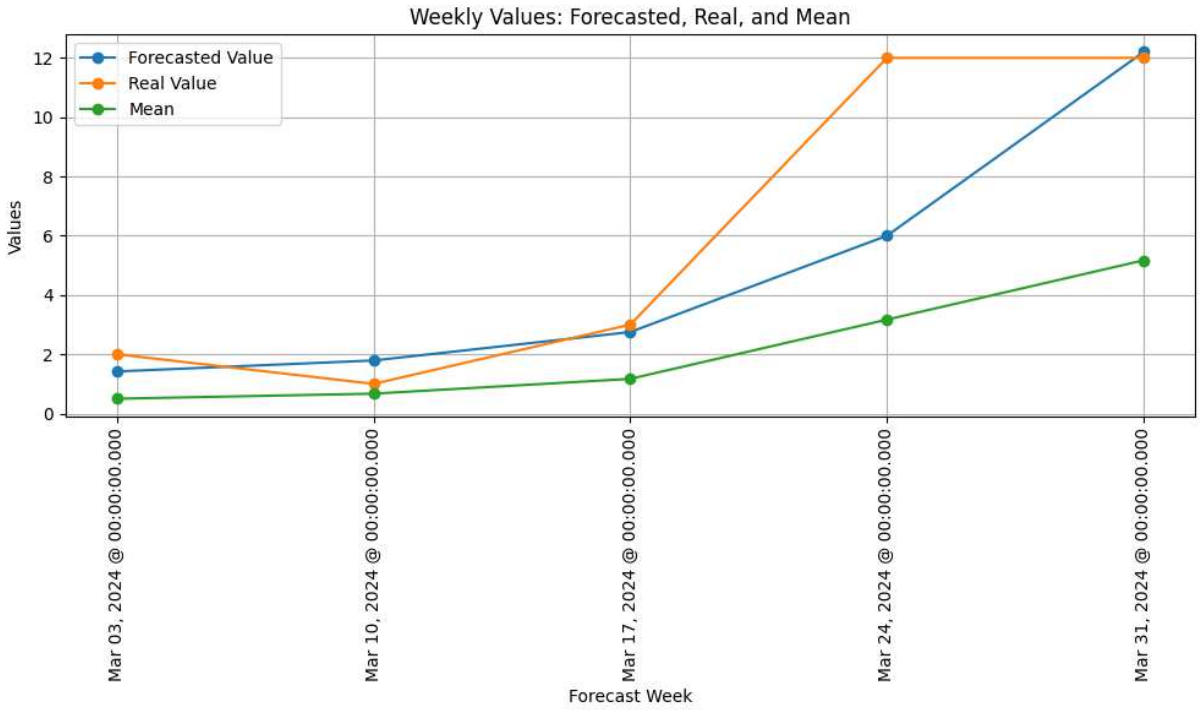

Figure 10 – Comparison Between Forecasted, Real, and Mean Values Over the Weeks in PoP_x18

- **Forecasted Values:** Represented in blue, these correspond to the model's predictions.
- **Real Values:** Represented in orange, these correspond to the observed values in reality.
- **Mean:** Represented in green, this reflects the average values over the period.

By analyzing Figure 10, it is possible to clearly observe the differences between the forecasts and the actual values over the weeks. Specifically:

- In the week starting on March 17, the forecast was close to the actual value (2.75 versus 3 incidents).
- In the week starting on March 24, the discrepancy was significant (6 forecasted versus 12 actual).

These analyses are essential for identifying error patterns and, consequently, adjusting the model to better represent reality in future scenarios. The graphical tool facilitates the visualization of these trends, enabling a more intuitive and objective evaluation of the anomalies between forecasts and actual data.

To better understand the 100% discrepancy between the actual malware incident values and the predicted values, the 12 malware incidents recorded during the period starting on March 24 were thoroughly analyzed using data from the RNP database.Table 12 provides a detailed analysis of all infected hosts in PoP_x18 for the week starting from March 24 to March 30. Analyzing the characteristics of the identified malware, a direct connection to the Endgame operation [78] and [79] is inferred, based on the types of malware and the dates on which they occurred. It is worth noting that the data presented in Table 12 were anonymized to protect the information of the institutions involved. Operation Endgame is an international initiative aimed at combating botnets. It employs techniques such as sinkholing, which redirects malicious traffic to servers controlled by authorities, allowing for the monitoring and analysis of botnet activities. Additionally, authorities notify security teams about infected servers and/or endpoints, enabling these teams to intervene in assets that were previously unknown to be compromised by malware. The malware notifications were provided by an important partner of RNP, internationally recognized for its efforts in combating botnets.

Table 12 – Record of infected hosts in PoP_x18 in the anomalous period analyzed

Date	Institution	Source IP	Protocol	Source Port	Malware Name	Destination IP	Destination Port
March 24, 2024	INSTITUTION 1	SOURCE IP 1	tcp	33452	andromeda	DST IP 1	80
March 25, 2024	INSTITUTION 2	SOURCE IP 2	tcp	65342	trickbot	DST IP 2	443
March 25, 2024	INSTITUTION 3	SOURCE IP 3	tcp	54740	smokeloader	DST IP 3	443
March 26, 2024	INSTITUTION 4	SOURCE IP 4	tcp	62490	trickbot	DST IP 2	443
March 26, 2024	INSTITUTION 5	SOURCE IP 5	tcp	38613	bokbot	DST IP 4	443
March 26, 2024	INSTITUTION 6	SOURCE IP 6	tcp	53013	adload	DST IP 5	443
March 27, 2024	INSTITUTION 1	SOURCE IP 7	tcp	63150	systembc	DST IP 6	443
March 27, 2024	INSTITUTION 7	SOURCE IP 8	tcp	49316	bumblebee	DST IP 7	443
March 28, 2024	INSTITUTION 2	SOURCE IP 9	tcp	49174	pikabot	DST IP 8	443
March 28, 2024	INSTITUTION 2	SOURCE IP 2	tcp	65347	trickbot	DST IP 2	443
March 29, 2024	INSTITUTION 9	SOURCE IP 10	tcp	54383	smokeloader	DST IP 3	443
March 29, 2024	INSTITUTION 1	SOURCE IP 1	tcp	54740	smokeloader	DST IP 3	443

Finally, the implemented visualization tool proved to be highly effective in identifying anomalous situations. When analyzing the actual and predicted values in Table 11, it is observed that the LSTM model consistently produced predictions close to the actual values, except for the week starting on March 24, when a significant divergence occurred. This variation, highlighted by the visualization tool, could have served as an early warning, indicating a potential anomaly and allowing the SOC team to use the tool to make proactive strategic decisions. The key differentiator of our method lies in the combination of the core activity of incident forecasting using the LSTM model with

the data visualization tool. This integration not only enables precise anomaly detection but also presents the results clearly and intuitively, facilitating quick interpretation and effective action by analysts.

Case 2: The second case addresses a situation recorded at PoP_x06. The data presented in Table 13 reflect a specific snapshot from a more extended period, covering from July 1, 2023, to July 1, 2024. This snapshot focuses on the interval between February 18 to April 21, highlighting weeks in which significant discrepancies were observed between the actual number of incidents and the predictions generated by the model, suggesting the presence of anomalies. The situations addressed in this case share the same characteristics as those in Case 1, specifically involving discrepancies where the difference between the predictions and actual values was equal to or greater than 30%, and the actual number of incidents was five or more. Additionally, the columns and their corresponding meanings in Table 13 are identical to those in Table 11.

Table 13 – Comparison of Predicted and Actual Weekly Malware Incident Values at PoP_x06

Institution	Week Start	Week Final	Forecast Week	Real Value	Forecasted Value	Mean
PoP_x06	Feb 18, 2024	Mar 24, 2024	Mar 31, 2024	13	19.77	9.5
PoP_x06	Feb 25, 2024	Mar 31, 2024	Apr 7, 2024	8	12.75	10.83
PoP_x06	Mar 3, 2024	Apr 7, 2024	Apr 14, 2024	20	18.41	14.17
PoP_x06	Mar 10, 2024	Apr 14, 2024	Apr 21, 2024	20	22.13	17.5
PoP_x06	Mar 17, 2024	Apr 21, 2024	Apr 28, 2024	18	22.85	16.83

Based on the data presented in Table 13, significant discrepancies are observed between the LSTM model's predictions and the actual malware incident values. For the period from February 18 to March 24, 2024, the model predicted 19.77 incidents for the week beginning on March 31, while the actual number was only 13, indicating a notable deviation. Similarly, for the period from February 25 to March 31, the model forecasted 12.75 incidents for the week of April 7, 2024, whereas the actual number recorded was 8. These variations, including one as high as 60%, underscore the challenges in accurately modeling such dynamic data. The visualization tool we developed, shown in Figure 11, provides further clarity by graphically representing these discrepancies, offering an accessible way to compare predicted and actual values.

Figure 11 – Comparison Between Forecasted, Real, and Mean Values Over the Weeks in PoP_x06

To delve deeper into the patterns revealed in the analysis, an investigation was conducted into the malware incidents during these periods, focusing particularly on the week starting March 31. As detailed in Table 14, these incidents were distributed across six institutions, with one accounting for six cases. Various types of malware, such as socks5systemz and avalanche-andromeda, were identified, and some IP addresses were affected by multiple malware types. This diversity in malware and infection patterns suggests a range of attack strategies and the exploitation of different system vulnerabilities. To gain a comprehensive understanding of these incidents, a detailed examination of the security configurations and vulnerabilities within the affected institutions is essential. This nuanced approach can help uncover underlying issues and inform better predictive and preventive measures in the future.

Table 14 – Record of infected hosts at PoP_x06 in the first anomalous period analyzed

Date	Institution	Source IP	Protocol	Source Port	Malware Name	Destination IP	Destination Port
March 31, 2024	INSTITUTION 1	SOURCE IP 1	tcp	20824	avalanche-andromeda	DESTINATION IP 1	80
April 01, 2024	INSTITUTION 2	SOURCE IP 2	tcp	50578	socks5systemz	DESTINATION IP 2	80
April 01, 2024	INSTITUTION 3	SOURCE IP 3	tcp	50578	pseudomanuscrypt	DESTINATION IP 3	443
April 01, 2024	INSTITUTION 2	SOURCE IP 4	tcp	50578	socks5systemz	DESTINATION IP 2	80
April 01, 2024	INSTITUTION 1	SOURCE IP 5	tcp	26629	avalanche-andromeda	DESTINATION IP 1	80
April 02, 2024	INSTITUTION 1	SOURCE IP 6	tcp	2345	socks5systemz	DESTINATION IP 2	80
April 02, 2024	INSTITUTION 1	SOURCE IP 6	tcp	2345	socks5systemz	DESTINATION IP 2	80
April 02, 2024	INSTITUTION 4	SOURCE IP 7	tcp	18295	pseudomanuscrypt	DESTINATION IP 3	443
April 03, 2024	INSTITUTION 5	SOURCE IP 8	tcp	2559	socks5systemz	DESTINATION IP 2	80
April 03, 2024	INSTITUTION 6	SOURCE IP 9	tcp	49188	avalanche-andromeda	DESTINATION IP 1	80
April 04, 2024	INSTITUTION 1	SOURCE IP 1	tcp	49188	pseudomanuscrypt	DESTINATION IP 3	443
April 04, 2024	INSTITUTION 2	SOURCE IP 2	tcp	49188	socks5systemz	DESTINATION IP 2	80
April 04, 2024	INSTITUTION 1	SOURCE IP 1	tcp	2559	socks5systemz	DESTINATION IP 2	80

Subsequently, the following period, the week starting on April 7, was analyzed, with the details of the registered malware presented in Table 15. During this period, one of the most recurrent malware types was socks5systemz, which appeared in two incidents, both involving the same destination IP, "DESTINATION IP 2," but using different source ports. This behavior suggests an attempt to vary attack approaches in order to evade detection, potentially indicating a coordinated campaign focused on a specific target over time.

Table 15 – Record of infected hosts at PoP_x06 in the second anomalous period analyzed

Date	Institution	Source IP	Protocol	Source Port	Malware Name	Destination IP	Destination Port
April 08, 2024	INSTITUTION 2	SOURCE IP 4	tcp	50578	socks5systemz	DESTINATION IP 2	80
April 08, 2024	INSTITUTION 1	SOURCE IP 5	tcp	2345	socks5systemz	DESTINATION IP 2	80
April 08, 2024	INSTITUTION 4	SOURCE IP 10	tcp	2133	pseudomanuscrypt	DESTINATION IP 3	443
April 10, 2024	INSTITUTION 3	SOURCE IP 11	tcp	53789	avalanche-andromeda	DESTINATION IP 1	80
April 10, 2024	INSTITUTION 6	SOURCE IP 12	tcp	56988	avalanche-andromeda	DESTINATION IP 1	80
April 11, 2024	INSTITUTION 2	SOURCE IP 13	tcp	49231	pseudomanuscrypt	DESTINATION IP 3	443
April 12, 2024	INSTITUTION 7	SOURCE IP 14	tcp	18944	adload	DESTINATION IP 4	443
April 12, 2024	INSTITUTION 8	SOURCE IP 15	tcp	43369	trickboat	DESTINATION IP 5	443

Finally, by comparing Tables 14 and 15, it is possible to identify recurring patterns in the malware incidents during both analyzed periods. In Table 14, the socks5systemz malware appears repeatedly, establishing communication with the destination IP "DESTI-

NATION IP 2" through different source ports, indicating a persistence in attacks focused on the same target. This behavior continues in Table 15, where socks5systemz reappears, still targeting the same destination IP but with fewer occurrences, suggesting a modification in the attackers' activities. Similarly, the avalanche-andromeda malware is also present in both periods, affecting several institutions, with a slight decrease in occurrences during the second week. The reduction in the total number of incidents, from 13 in the first period to 8 in the second, may be attributed to several factors. It is possible that the institutions adopted more effective security measures, mitigating some of the vulnerabilities exploited in the previous week. Another hypothesis is that the attackers adjusted their strategies, redirecting their efforts to new targets or adapting their tactics in response to more robust defenses. However, despite the detailed analysis of the types of malware present in each period, it was not possible to establish any direct relationship between these malware incidents and specific coordinated malware campaigns during the analyzed period. This indicates that, although the same malware appeared in multiple institutions, the observed patterns are not conclusive enough to infer the existence of a structured campaign.

In summary, in case 2, the visualization tool proved effective in identifying anomalous situations, as evidenced in Table 13. In the first period, corresponding to the week beginning on March 31, the LSTM model predicted 19.77 incidents, while the actual number was 13. In the second period, beginning on April 7, the prediction was 12.75 incidents, and the actual recorded number was 8, reflecting a downward trend. In case 1, the observed discrepancy involved situations where the prediction was lower than the actual number of incidents. However, in the second case analyzed, the opposite occurred: the actual number of incidents was lower than the predicted values. This difference was identified with the support of the visualization tool, thus highlighting its versatility in data analysis.

Case 3: The third study addresses a situation that occurred at PoP_x19. Table 16 presents a specific excerpt from a broader period, spanning from July 1, 2023, to July 1, 2024. For this case, the analyzed period focuses on March 17 to May 26, 2024, with an emphasis on the week in which a significant discrepancy was detected between actual malware incidents and the model's predictions, indicating possible anomalies. As in Cases 1 and 2, the situations examined here involve differences of 30% or more between predicted and observed values, with the actual number of incidents being five or more. Additionally, the columns and their respective meanings in Table 16 maintain the same format as in the previous cases, ensuring data structure consistency and facilitating comparative analysis.

Table 16 – Comparison of Predicted and Actual Weekly Malware Incident Values at PoP_x19

Institution	Week Start	Week Final	Forecast Week	Real Value	Forecasted Value	Mean
PoP_x19	Mar 17, 2024	Apr 21, 2024	Apr 28, 2024	15	16.09	16
PoP_x19	Mar 24, 2024	Apr 28, 2024	May 5, 2024	13	13.15	14.33
PoP_x19	Mar 31, 2024	May 5, 2024	May 12, 2024	15	14.94	14
PoP_x19	Apr 7, 2024	May 12, 2024	May 19, 2024	11	11.98	12.83
PoP_x19	Apr 14, 2024	May 19, 2024	May 26, 2024	6	10.61	12.17
PoP_x19	Apr 21, 2024	May 26, 2024	Jun 2, 2024	13	13.95	12.17

The analysis of Table 16 highlights notable discrepancies between the LSTM model's predictions and the actual number of malware incidents, particularly during the period from April 14 to May 19, 2024. This divergence becomes most apparent in the week starting May 26, when the model predicted 11.94 incidents, but only 6 incidents were recorded—a difference of nearly 50%. Such a significant variation underscores a marked decline in incidents compared to the forecast, drawing attention to the need for enhanced contextual analysis. To aid in understanding these discrepancies, the visualization tool shown in Figure 12 offers a graphical representation of the data, facilitating quick comparisons and trend identification without relying solely on tabular information.

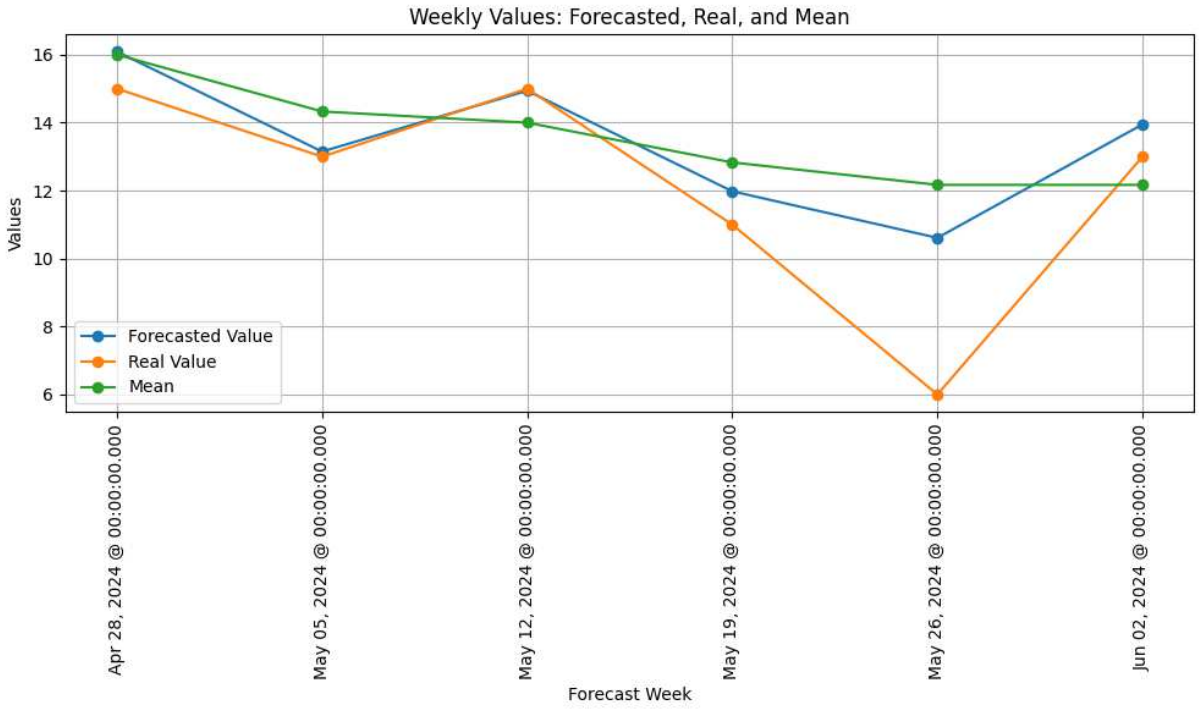

Figure 12 – Comparison Between Forecasted, Real, and Mean Values Over the Weeks in PoP_x19

Building on this understanding, a more detailed examination of malware dynamics was undertaken for the week beginning May 26, focusing on the preceding week starting May 19 to identify potential underlying causes. This retrospective investigation is critical for contrasting periods with minimal discrepancies against those with significant divergences. Table 17 presents detailed information on the malware incidents during the week of May 19, categorizing them by type and institutional distribution. Seven distinct institutions experienced infections, with repeated occurrences of malware such as pseudomanuscrypt, which contributed significantly to the incident count. The diversity of threats—including widely recognized malware like avalanche-andromeda and socks5systemz, as well as more advanced variants like likely-rat-firebird—highlights the range of vulnerabilities exploited. These findings emphasize the necessity of investigating security configurations and response strategies within the affected institutions to address both recurring and novel threats effectively.

Table 17 – Record of infected hosts in PoP_x19 in the first anomalous period analyzed

Date	Institution	Source IP	Protocol	Source Port	Malware Name	Destination IP	Destination Port
May 20, 2024	INSTITUTION 1	SOURCE IP 1	tcp	39340	avalanche-andromeda	DST IP 1	80
May 20, 2024	INSTITUTION 2	SOURCE IP 2	tcp	52849	likely-rat-firebird	DST IP 2	443
May 21, 2024	INSTITUTION 3	SOURCE IP 3	tcp	29613	adload	DST IP 3	443
May 21, 2024	INSTITUTION 1	SOURCE IP 1	tcp	38760	avalanche-andromeda	DST IP 1	80
May 22, 2024	INSTITUTION 4	SOURCE IP 4	tcp	60999	pseudomanuscrypt	DST IP 4	443
May 23, 2024	INSTITUTION 5	SOURCE IP 5	tcp	57222	pseudomanuscrypt	DST IP 4	443
May 23, 2024	INSTITUTION 6	SOURCE IP 6	tcp	36402	bumblebee	DST IP 5	443
May 23, 2024	INSTITUTION 7	SOURCE IP 7	tcp	40341	downadup	DST IP 6	80
May 24, 2024	INSTITUTION 2	SOURCE IP 2	tcp	53164	socks5systemz	DST IP 7	80
May 24, 2024	INSTITUTION 4	SOURCE IP 4	tcp	57852	pseudomanuscrypt	DST IP 4	443
May 24, 2024	INSTITUTION 5	SOURCE IP 5	tcp	54009	pseudomanuscrypt	DST IP 4	443

Continuing the analysis, during the week beginning on May 26, 2024, a particularly relevant period due to the significant discrepancy observed—six malware incidents were recorded, as detailed in Table 18. During this interval, the pseudomanuscrypt malware was the most recurrent, appearing in three incidents, affecting multiple institutions, and establishing communication with the same destination IP, "DST IP 4", though using different source ports. This variation in ports suggests a deliberate attempt to bypass detection systems, indicating a coordinated attack focused on a specific target. Additionally, other malware such as socks5systemz, avalanche-andromeda, and adload were also identified. The variation in port usage and the persistence in targeting the same institutions suggest that attackers are employing multiple tactics to evade detection, reinforcing the need for a more effective response to mitigate these coordinated threats.

Table 18 – Record of infected hosts in PoP_x19 in the second anomalous period analyzed

Date	Institution	Source IP	Protocol	Source Port	Malware Name	Destination IP	Destination Port
May 26, 2024	INSTITUTION 2	SOURCE IP 2	tcp	510235	adload	DST IP 3	443
May 27, 2024	INSTITUTION 4	SOURCE IP 4	tcp	60587	pseudomanuscrypt	DST IP 4	443
May 27, 2024	INSTITUTION 2	SOURCE IP 2	tcp	53164	socks5systemz	DST IP 7	80
May 29, 2024	INSTITUTION 5	SOURCE IP 5	tcp	54009	pseudomanuscrypt	DST IP 4	443
May 29, 2024	INSTITUTION 8	SOURCE IP 8	tcp	60999	pseudomanuscrypt	DST IP 4	443
May 31, 2024	INSTITUTION 9	SOURCE IP 9	tcp	57222	avalanche-andromeda	DST IP 1	80

By comparing the malware incidents in Tables 17 and 18, recurring patterns in malware incidents can be identified across the analyzed periods. In Table 17, the pseudomanuscrypt malware is particularly prevalent, affecting multiple institutions and using the same destination IP, "DST IP 4", for communication, but through different source ports. This pattern of repeated attacks targeting the same destination IP suggests a possible coordinated campaign aimed at exploiting specific vulnerabilities. This behavior persists in Table 18, where pseudomanuscrypt continues to be one of the most recurrent malware, still targeting the same IP, "DST IP 4". The persistence of the attack over two periods, with variations in the institutions and source ports, reinforces the hypothesis that the attackers are focused on exploiting a specific target, attempting to evade detection by altering the communication ports.

The reduction in the total number of malware incidents between the two periods is noteworthy, decreasing from 11 recorded incidents to 6. This decline could be attributed to more effective security measures implemented by the affected institutions. Another possibility is that the attackers adjusted their strategies, redirecting their efforts toward new targets. Despite the detailed analysis of the malware present in each period, it was not possible to establish a direct link between the attacks and a structured malware campaign, even though the same malware types appeared across different institutions. The observed patterns do not provide conclusive evidence to infer the existence of a centralized coordination behind the attacks, suggesting that the variations in occurrences might be the result of random attempts to exploit vulnerabilities. In Case 3, once again, we observed a discrepancy similar to that found in Case 2, where the number of incidents predicted by the model was significantly higher than the actual occurrences. As in Case 2, this difference could only be clearly identified using the visualization tool, which proved crucial in detecting the discrepancy.

Therefore, it can be concluded that the data visualization solution, an integral part of the proposed method, plays a crucial role in facilitating the identification of anomalous situations. This integration allows security analysts to detect anomalies more clearly and quickly, often enabling proactive actions. Additionally, the proposed method proves to be

an effective guide for assisting information security analysts in identifying future anomalies within their networks. The accuracy demonstrated throughout the various analyzed situations highlights its potential to enhance response capabilities to malware attacks. This efficiency makes the method suitable for use in Security Information and Event Management (SIEM) tools, with the metric serving as a threshold that triggers automatic alerts if the number of incidentes deviate from the LSTM-model predicted value. In this way, the proposed method not only enables a quick and effective response to information security incidents but also strengthens IT governance and management, helping to mitigate risks and ensure service continuity.

Not all organizations will be able to implement the solution with the same level of effectiveness though. A minimum level of operational maturity is required to ensure the accuracy of the method. This level of maturity is established according to the mandatory requirements for FIRST membership, which are based on the SIM3 model. Therefore, security teams that meet these criteria, as described in [80], are more likely to replicate the results presented. Organizations with lower maturity levels can still benefit from the method, but the results may vary significantly from those outlined in this work.

Moreover, the research revealed that the method performs significantly better in environments where there is a regularity of incidents. As evidenced in Table 2, the best results were observed in PoPs with a weekly average of incidents exceeding 40. This suggests that an increase in the frequency of incidents tends to improve the accuracy of predictions. Regarding the training period, the most consistent results were obtained with intervals equal to or greater than two years, which may limit the applicability of the method in organizations that are just beginning to monitor their malware incidents. For these institutions, the use of already developed and tested neural network models is recommended, with parameter adjustments to optimize prediction quality.

6 CONCLUSION

This work proposes the development of a method to predict malware incidents using a neural network model that analyzes time series with the LSTM technique. The method includes a visualization tool that allows security analysts to examine trends and behavioral variations in detail, facilitating the identification of patterns and anomalies.

To evaluate its feasibility, this study examined the effectiveness of LSTM models in predicting malware incident rates at public higher education institutions in Brazil, dividing the analysis into two distinct scenarios for a detailed comparison. The experiments in Scenario 1 used a dataset composed of real incident data from 10 Brazilian universities. The selection of the institutions was based on their scale, considering the number of public IP addresses and the volume of incidents.

The results indicate that larger institutions, characterized by a higher volume of incidents and less variable behavior, tend to have lower prediction errors. In contrast, smaller institutions show average errors exceeding 100% of the incident rate. The results did not reveal a clear correlation between the number of weeks used as input and the forecast accuracy for both smaller and larger institutions. As for the time period used for training, different results emerged for smaller and larger institutions. Smaller institutions did not show a clear relationship between the training period and the accuracy of the models. On the other hand, larger institutions achieved slightly better results when the training series spanned 2 years of data. Finally, the error rates for LSTM models were compared with those obtained with ARIMA models. LSTM outperformed ARIMA models by a modest margin, achieving smaller errors by 10%. Subsequently, to determine if the behavior of Scenario 1 could be expanded to a larger environment, exactly the same tests were conducted for this much larger new scenario.

In Scenario 2, the results continued to show that there is no clear correlation between the number of weeks used as input and the accuracy of the forecast. However, the six-week period stood out as the most effective, suggesting that adjusting the number of weeks of data used as input might be a promising strategy for improving the accuracy of predictions. As for the time period used for training, just as in the previous scenario where longer periods resulted in fewer errors, in this case, the best results occurred when the training series spanned 3 years of data. Lastly, we again compared the error rates for LSTM models with those obtained with ARIMA models. LSTM consistently outperformed ARIMA models by a significant margin, achieving lower errors by 40% on average.

Finally, new tests were conducted with the primary objective of evaluating how the data visualization tool, an essential part of the proposed method, could provide additional

benefits to the method. It is important to emphasize that the core of the method, the neural network model based on the LSTM technique for time series analysis, had already been previously validated and proven its value in identifying anomalies and predicting incidents. These new tests, therefore, focused on assessing how the visualization tool could further enhance the method. The tool proved to be effective in enabling security analysts to quickly and accurately identify anomalies between actual and predicted incidents. However, it should be noted that the method's effectiveness depends significantly on the maturity level of the organizations implementing it, being more suitable for teams that meet the minimum requirements for FIRST membership.

As future work, it is proposed to investigate the behavior of the LSTM network by adding more hidden layers. Furthermore, it is suggested to explore other architectures of recurrent neural networks, such as GRU networks and Transformer memory networks. It is also proposed to analyze other variables related to malware incidents, such as malware type and attack origins, to enrich the model and obtain more comprehensive insights.

BIBLIOGRAPHY

[1] National Institute of Standards and Technology (NIST). *Computer Security Incident Handling Guide*. Gaithersburg, 2012. Available at: <https://nvlpubs.nist.gov/nistpubs/specialpublications/nist.sp.800-61r2.pdf>. Accessed on: 09 Mar. 2024.

[2] AV-TEST Institute. *Malware Statistics & Trends Report*. 2024. Available at: <https://www.av-test.org/en/statistics/malware/>. Accessed on: 05 Jan. 2024.

[3] NOVINSON, M. *The 10 Biggest Ransomware Attacks of 2019*. 2019. Available at: <https://www.crn.com/slide-shows/security/the-10-biggest-ransomware-attacks-of-2019/>. Accessed on: 05 Jan. 2024.

[4] POINT, F. *Conti Ransomware: Inside One of the World's Most Aggressive Ransomware Groups*. 2022. Available at: <https://flashpoint.io/blog/history-of-conti-ransomware/>. Accessed on: 05 Jan. 2024.

[5] KASPERSKY. *What is Maze Ransomware? Definition and Explanation by Kaspersky*. n.d. Available at: <https://www.kaspersky.com/resource-center/definitions/what-is-maze-ransomware/>. Accessed on: 05 Jan. 2024.

[6] MICRO, T. *Ransomware Spotlight: REvil*. 2021. Available at: <https://www.trendmicro.com/vinfo/us/security/news/ransomware-spotlight/ransomware-spotlight-revil>. Accessed on: 05 Jan. 2024.

[7] ONE, S. *The Good, The Bad and the Ugly in Cybersecurity - Week 21*. 2022. Available at: <https://www.sentinelone.com/blog/the-good-the-bad-and-the-ugly-in-cybersecurity-week-21-3/>. Accessed on: 05 Jan. 2024.

[8] CISA. *Understanding Ransomware Threat Actors: LockBit | CISA*. 2023. Available at: <https://www.cisa.gov/news-events/cybersecurity-advisories/aa23-165a>. Accessed on: 05 Jan. 2024.

[9] KASPERSKY. *New BlackCat Ransomware*. 2022. Available at: <https://www.kaspersky.com/blog/black-cat-ransomware/44120/>. Accessed on: 05 Jan. 2024.

[10] AD, C. *2021 Top Routinely Exploited Vulnerabilities*. 2022. Available at: <https://www.cisa.gov/news-events/cybersecurity-advisories/aa22-117a>. Accessed on: 05 Jan. 2024.

[11] POINT, C. *Check Point Research Reports a 38% Increase in 2022 Global Cyberattacks*. 2023. Available at: <https://blog.checkpoint.com/2023/01/05/38-increase-in-2022-global-cyberattacks/>. Accessed on: 05 Jan. 2024.

[12] LALLIE, H. S. et al. Understanding cyber threats against the universities, colleges, and schools. *arXiv preprint arXiv:2307.07755*, 2023. Available at: <https://arxiv.org/abs/2307.07755>. Accessed on: 12 Jul. 2024.

[13] ISACA. *COBIT 5*. 2023. Available at: <https://www.isaca.org/-/media/files/isacadp/feature/documents/private/nonmember/related-items/cobit5-tool-kit.zip?la=en&hash=D29695FF113281575D1415F6FD80C59A77292703>. Accessed on: 20 Jul. 2024.

[14] AXELOS. *ITIL v4*. 2023. Available at: <https://www.axelos.com/resource-hub/practice/readers-manual-itil-4-practice-guide>. Accessed on: 20 Jul. 2024.

[15] ISO. *Cyber Security Incident - ISO 27000*. 2018. Available at: <https://www.iso.org/obp/ui/#iso:std:iso-iec:27000:ed-5:v1:en>. Accessed on: 09 Mar. 2024.

[16] RNP. *Cyber Security Incident - RNP*. 2019. Available at: <https://www.rnp.br/sistema-rnp/cais/rfc2350>. Accessed on: 03 Mar. 2024.

[17] CERT-BR. *Cyber Security Incident - CERT.br*. 2020. Available at: <https://www.cert.br/sobre/>. Accessed on: 03 Mar. 2024.

[18] ENISA. *Cyber Security Incident - ENISA*. 2021. Available at: <https://www.enisa.europa.eu/news/enisa-news/when-how-to-report-security-incidents>. Accessed on: 03 Mar. 2024.

[19] FIRST. *Cyber Security Taxonomies - FIRST*. 2018. Available at: <https://www.first.org/resources/papers/hamburg2018/ReferenceTaxonomy-FIRST.pdf>. Accessed on: 09 Mar. 2024.

[20] CIRCL. *Cyber Security Taxonomies - CIRCL*. 2018. Available at: <https://www.circl.lu/pub/taxonomy/>. Accessed on: 09 Mar. 2024.

[21] AGRAFIOTIS, I. et al. A taxonomy of cyber-harms: Defining the impacts of cyber-attacks and understanding how they propagate. *Journal of Cybersecurity*, v. 4, n. 1, p. tyy006, 10 2018. ISSN 2057-2085. Available at: <https://doi.org/10.1093/cybsec/tyy006>.Accessed on: 11 Jul. 2024.

[22] MENGES, F.; PERNUL, G. A comparative analysis of incident reporting formats. *Computers & Security*, Elsevier, v. 73, p. 87–101, 2018. Available at: <https://www.sciencedirect.com/science/article/pii/S0167404817302250>. Accessed on: 11 Jul. 2024.

[23] IETF. *Cyber Security Reports - IETF*. 2016. Available at: <https://datatracker.ietf.org/doc/rfc7970/>. Accessed on: 09 Mar. 2024.

[24] COLOME, M.; NUNES, R.; SILVA, L. Case-based cybersecurity incident resolution. In: *Proceedings of the 31st International Conference on Software Engineering and Knowledge Engineering*. [S.l.: s.n.], 2019. p. 253–258. Available at: <https://doi.org/10.18293/SEKE2019-204>. Accessed on: 09 Mar. 2024.

[25] TUFTE, S. E. Documenting cyber security incidents. *Department of Management Science and Engineering, Stanford University, School of Information, UC Berkeley*, 2015. Available at: <https://fsi9-prod.s3.us-west-1.amazonaws.com/s3fs-public/documenting_cyber_incidents.pdf>. Accessed on: 09 Mar. 2024.

[26] SANS. *Incident Response Cycle*. 2016. Available at: <https://www.sans.org/media/score/504-incident-response-cycle.pdf>. Accessed on: 13 Mar. 2024.

[27] BRASILEIRO, P. *National Information Security Police*. 2018. Available at: <https://www.planalto.gov.br/ccivil_03/_ato2015-2018/2018/decreto/D9637.htm>. Accessed on: 09 Mar. 2024.

[28] BRASILEIRO, P. *National Information Security Police*. 2020. Available at: <https://www.planalto.gov.br/ccivil_03/_Ato2019-2022/2020/Decreto/D10222.htm>. Accessed on: 09 Mar. 2024.

[29] BRASILEIRO, P. *Lei Geral de Proteção de Dados (LGPD)*. 2018. Available at: <https://www.planalto.gov.br/ccivil_03/_ato20152018/2018/lei/l13709.htm>. Accessed on: 09 Mar. 2024.

[30] CHRISTODORESCU, M. et al. Semantics-aware malware detection. In: *Proceedings of the 2005 IEEE Symposium on Security and Privacy*. [S.l.: s.n.], 2005. p. 32–46. ISBN 0-7695-2339-0. Available at: <https://doi.org/10.1109/SP.2005.20>. Accessed on: 03 Apr. 2024.

[31] MCGRAW, G.; MORRISETT, G. Attacking malicious code: A report to the infosec research council. *Software, IEEE*, v. 17, p. 33–41, out. 2000. Available at: <https://ieeexplore.ieee.org/document/6156709>. Accessed on: 03 Apr. 2024.

[32] SOUPPAYA, M.; SCARFONE, K. et al. Guide to malware incident prevention and handling for desktops and laptops. *NIST Special Publication*, v. 800, p. 83, 2013. Available at: <https://csrc.nist.gov/pubs/sp/800/83/r1/final>. Accessed on: 03 Apr. 2024.

[33] WANGEN, G. The role of malware in reported cyber espionage: A review of the impact and mechanism. *Information*, v. 6, n. 2, p. 183–211, 2015. ISSN 2078-2489. Available at: <https://www.mdpi.com/2078-2489/6/2/183/pdf?version=1432042905>. Accessed on: 03 Apr. 2024.

[34] GUO, H.; CHENG, H. K.; KELLEY, K. Impact of network structure on malware propagation: A growth curve perspective. *Journal of Management Information Systems*, Taylor & Francis, v. 33, n. 1, p. 296–325, 2016. Available at: <https://www3.nd.edu/~kkelley/publications/articles/Guo_Cheng_Kelley_JMIS_2016.pdf>. Accessed on: 03 Apr. 2024.

[35] KAUR, J. Taxonomy of malware: virus, worms and trojan. *Int. J. Res. Anal. Rev*, v. 6, n. 1, p. 192–196, 2019. Available at: <https://ijrar.com/upload_issue/ijrar_issue_20544162.pdf>. Accessed on: 03 Apr. 2024.

[36] VINOD, P. et al. Survey on malware detection methods. In: *Proceedings of the 3rd Hackers' Workshop on Computer and Internet Security (IITKHACK'09)*. [S.l.: s.n.], 2009. p. 74–79. Available at: <https://d1wqtxts1xzle7.cloudfront.net/73571775/proceedings_hack.in-libre.pdf?1635141648=&response-content-disposition=inline%3B+filename%3DA_Hardware_Authentication_Architecture_f.pdf&Expires=1732388972&Signature=PVdqAiDTYq~HqTlt8RJLpCTWpkbYg3NmCu5Be~U7qkcKiuI6Fk5htocRdD8-Lh88Q4k-qfJzs7

_&Key-Pair-Id=APKAJLOHF5GGSLRBV4ZA#page=82>.Accessed on: 03 Apr. 2024.

[37] ALENEZI, M. et al. Evolution of malware threats and techniques: A review. *International Journal of Communication Networks and Information Security*, v. 12, p. 326, 12 2020. Available at: <https://www.ijcnis.org/index.php/ijcnis/article/view/4723/614>.Accessed on: 03 Apr. 2024.

[38] UNIFESP. *Unifesp Ransomware*. 2022. Available at: <https://www.defcon-lab.org/unifesp-vice-society-ransomware/>. Accessed on: 13 Mar. 2024.

[39] USP. *USP Ransomware*. 2023. Available at: <https://www.lgpdbrasil.com.br/hospital-universitario-da-usp-tem-atendimentos-afetados-por-ataque-hacker-que-paralisa-servic>.Accessed on: 13 Mar. 2024.

[40] GUTTMAN, E.; BROWNLEE, N. *Expectations for Computer Security Incident Response*. [S.l.]: RFC Editor, 1998. RFC 2350. (Request for Comments, 2350). Available at: <https://www.rfc-editor.org/info/rfc2350>. Accessed on: 13 Mar. 2024.

[41] CTIR. *CTIR Gov*. 2024. Available at: <https://www.gov.br/ctir/pt-br>. Accessed on: 17 Mar. 2024.

[42] FIRST. *FIRST CSIRT Services*. 2017. Available at: <https://www.first.org/resources/guides/Establishing-CSIRT-v1.2.pdf>. Accessed on: 17 Mar. 2024.

[43] ENISA. *ENISA CSIRT Services*. 2020. Available at: <https://www.enisa.europa.eu/publications/how-to-set-up-csirt-and-soc/@@download/fullReport>. Accessed on: 17 Mar. 2024.

[44] CIRCL. *CIRCL CSIRT Members*. 2024. Available at: <https://www.circl.lu/team/>. Accessed on: 17 Mar. 2024.

[45] KASSIM, S. R. B. M.; LI, S.; ARIEF, B. Incident response practices across national csirts: Results from an online survey. *OIC-CERT Journal of Cyber Security*, OIC CERT, v. 4, n. 1, p. 67–84, 2022. Available at: <https://kar.kent.ac.uk/94119/1/Incident-Response-Practice.pdf>. Accessed on: 21 May 2024.

[46] VACHEK, P. Audit system at cesnet-certs. *Target*, v. 192, p. 123–145, 2009. Available at: <https://dl.acm.org/doi/10.5555/1718109.1718113>. Accessed on: 21 May 2024.

[47] AKSU, M. U.; ALTUNCU, E.; BICAKCI, K. A first look at the usability of openvas vulnerability scanner. In: *Workshop on Usable Security (USEC)*. [S.l.: s.n.], 2019. Available at: <https://www.ndss-symposium.org/wp-content/uploads/2019/02/usec2019_03-4_Aksu_paper.pdf>. Accessed on: 21 May 2024.

[48] DOMBAJOVá, A.-M. Tool for persistent scanning of network vulnerabilities. *Master Degree*, 2022. Available at: <https://is.muni.cz/th/a8f4f/Tool_for_persistent_scanning_of_network_vulnerabilities.pdf>. Accessed on: 06 Jul. 2024.

[49] HENDRA, R.; HANITA, M. et al. The implementation of cyber incident management frameworks in indonesia. *Jurnal Teknologi Informasi dan Pendidikan*, v. 13, n. 2, p. 9–16, 2020. Available at: <https://scholar.ui.ac.id/en/publications/the-implementation-of-cyber-incident-management-frameworks-in-ind>. Accessed on: 06 Jul. 2024.

[50] OPENCSIRT. *OpenCSIRT*. 2023. Available at: <https://www.trusted-introducer.org/SIM3-mentoring.pdf>. Accessed on: 31 Mar. 2024.

[51] FIRST. *FIRST Membership Program*. 2023. Available at: <https://www.first.org/membership/process>. Accessed on: 17 Mar. 2024.

[52] MOHRI, M.; ROSTAMIZADEH, A.; TALWALKAR, A. *Foundations of Machine Learning*. 2. ed. Cambridge, MA: MIT Press, 2018. (Adaptive Computation and Machine Learning). Available at: <https://www.hlevkin.com/hlevkin/45MachineDeepLearning/ML/Foundations_of_Machine_Learning.pdf>. Accessed on: 17 Mar. 2024.

[53] NAEEM, S. et al. An unsupervised machine learning algorithms: Comprehensive review. *International Journal of Computing and Digital Systems*, University of Bahrain, 2023. Available at: <https://www.researchgate.net/profile/Aqib-Ali-6/publication/368983958_An_Unsupervised_Machine_Learning_Algorithms_Comprehensive_Review/links/643c783f1b8d044c632ba4ab/An-Unsupervised-Machine-Learning-Algorithms-Comprehensive-Review.pdf>. Accessed on: 17 Mar. 2024.

[54] FU, T. chung. A review on time series data mining. *Engineering Applications of Artificial Intelligence*, v. 24, n. 1, p. 164–181, 2011. ISSN 0952-1976. Available at: <https://doi.org/10.1016/j.engappai.2010.09.007>. Accessed on: 17 Mar. 2024.

[55] LIM, B.; ZOHREN, S. Time-series forecasting with deep learning: a survey. *Philosophical Transactions of the Royal Society A*, The Royal Society Publishing, v. 379, n. 2194, p. 20200209, 2021. Available at: <https://doi.org/10.1098/rsta.2020.0209>. Accessed on: 17 Mar. 2024.

[56] BAHETI, A.; TOSHNIWAL, D. Trend analysis of time series data using data mining techniques. In: *2014 IEEE International Congress on Big Data*. [S.l.: s.n.], 2014. p. 430–437. Available at: <https://ieeexplore.ieee.org/document/6906812>. Accessed on: 17 Mar. 2024.

[57] DAVEY, A.; FLORES, B. Identification of seasonality in time series: A note. *Mathematical and Computer Modelling*, v. 18, n. 6, p. 73–81, 1993. ISSN 0895-7177. Available at: <https://doi.org/10.1016/0895-7177(93)90126-J>. Accessed on: 17 Mar. 2024.

[58] KLEY, T.; PREUSS, P.; FRYZLEWICZ, P. Predictive, finite-sample model choice for time series under stationarity and non-stationarity. *Electronic Journal of Statistics*, Institute of Mathematical Statistics and Bernoulli Society, v. 13, n. 2, p. 3710–3774, 2019. Available at: <https://doi.org/10.1214/19-EJS1606>. Accessed on: 17 Mar. 2024.

[59] ELSARAITI, M.; MERABET, A. A comparative analysis of the arima and lstm predictive models and their effectiveness for predicting wind speed. *Energies*, MDPI, v. 14, n. 20, p. 6782, 2021. Available at: <https://www.mdpi.com/1996-1073/14/20/6782/pdf?version=1634631524>. Accessed on: 17 Mar. 2024.

[60] BOX, G. E. P. et al. *Time Series Analysis: Forecasting and Control*. [S.l.]: John Wiley & Sons, 2015. Available at: <https://onlinelibrary.wiley.com/doi/10.1111/jtsa.12194>. Accessed on: 03 Apr. 2024.

[61] GERS, F. A.; SCHMIDHUBER, J.; CUMMINS, F. Learning to forget: Continual prediction with lstm. *Neural Computation*, MIT Press, v. 12, n. 10, p. 2451–2471, 2000. Available at: <https://direct.mit.edu/neco/article-abstract/12/10/2451/6415/Learning-to-Forget-Continual-Prediction-with-LSTM?redirectedFrom=fulltext>. Accessed on: 03 Apr. 2024.

[62] HOCHREITER, S.; SCHMIDHUBER, J. Lstm can solve hard long time lag problems. *Advances in Neural Information Processing Systems*, v. 9, 1996. Available at: <https://proceedings.neurips.cc/paper_files/paper/1996/file/a4d2f0d23dcc84ce983ff9157f8b7f88-Paper.pdf>. Accessed on: 11 Apr. 2024.

[63] FANG, X. et al. A deep learning framework for predicting cyber attacks rates. *EURASIP Journal on Information Security*, Springer, v. 2019, p. 1–11, 2019. Available at: <https://jis-eurasipjournals.springeropen.com/counter/pdf/10.1186/s13635-019-0090-6.pdf>. Accessed on: 11 Apr. 2024.

[64] HUSáK, M. et al. Survey of attack projection, prediction, and forecasting in cyber security. *IEEE Communications Surveys & Tutorials*, v. 21, n. 1, p. 640–660, 2019. Available at: <https://ieeexplore.ieee.org/document/8470942>. Accessed on: 11 Apr. 2024.

[65] SHEN, Y. et al. Tiresias: Predicting security events through deep learning. In: *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. New York, NY, USA: Association for Computing Machinery, 2018. (CCS '18), p. 592–605. ISBN 9781450356930. Available at: <https://doi.org/10.1145/3243734.3243811>. Accessed on: 11 Apr. 2024.

[66] ALAHMADI, B. A.; AXON, L.; MARTINOVIC, I. 99% false positives: A qualitative study of SOC analysts' perspectives on security alarms. In: *31st USENIX Security Symposium (USENIX Security 22)*. Boston, MA: USENIX Association, 2022. p. 2783–2800. ISBN 978-1-939133-31-1. Available at: <https://www.usenix.org/conference/usenixsecurity22/presentation/alahmadi>.Accessed on: 11 Apr. 2024.

[67] BAKDASH, J. Z. et al. Malware in the future? Forecasting of analyst detection of cyber events. *Journal of Cybersecurity*, v. 4, n. 1, p. tyy007, 12 2018. ISSN 2057-2085. Available at: <https://academic.oup.com/cybersecurity/article-pdf/4/1/tyy007/27239107/tyy007.pdf>. Accessed on: 11 Apr. 2024.

[68] WERNER, G. et al. Forecasting cyberattacks as time series with different aggregation granularity. In: *2018 IEEE International Symposium on Technologies for Homeland Security (HST)*. [S.l.: s.n.], 2018. p. 1–7. Available at: <https://ieeexplore.ieee.org/document/8574185>. Accessed on: 15 Apr. 2024.

[69] MAHMOOD, S. H. A.; ABBASI, A. Using deep generative models to boost forecasting: A phishing prediction case study. In: *2020 International Conference on Data Mining Workshops (ICDMW)*. [S.l.: s.n.], 2020. p. 496–505. Available at: <https://ieeexplore.ieee.org/document/9346362>. Accessed on: 15 Apr. 2024.

[70] ZUZČÁK, M.; BUJOK, P. Using honeynet data and a time series to predict the number of cyber attacks. *Computer Science and Information Systems*, v. 18, n. 4, p. 1197–1217, 2021. Available at: <https://doi.org/10.2298/CSIS200715040Z>.Accessed on: 15 Apr. 2024.

[71] REIJERS, H. A. Business process management: The evolution of a discipline. *Computers in Industry*, v. 126, p. 103404, 2021. ISSN 0166-3615. Available at: <https://doi.org/10.1016/j.compind.2021.103404>. Acessed on: 15 Apr. 2024.

[72] ALZUBAIDI, L. et al. Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *Journal of big Data*, Springer, v. 8, p. 1–74, 2021. Available at: <https://journalofbigdata.springeropen.com/counter/pdf/10.1186/s40537-021-00444-8.pdf>. Acessed on: 15 Apr. 2024.

[73] GREFF, K. et al. Lstm: A search space odyssey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, IEEE, v. 39, n. 8, p. 1759–1775, 2017.

[74] SMITH, T. G. *pmdarima: ARIMA estimators for Python*. 2023. Available at: <https://alkaline-ml.com/pmdarima/>. Accessed on: 05 Jan. 2024.

[75] HAMEED, S. et al. Deep learning based multimodal urban air quality prediction and traffic analytics. *Scientific Reports*, Nature Publishing Group UK London, v. 13, n. 1, p. 22181, 2023. Available at: <https://www.nature.com/articles/s41598-023-49296-7.pdf>. Acessed on: 15 Apr. 2024.

[76] ROSSUM, G. V. *Python Tutorial*. 1995. Centrum voor Wiskunde en Informatica (CWI).

[77] HUNTER, J. D. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, v. 9, n. 3, p. 90–95, 2007.

[78] EUROPOL. *Europol Endgame*. 2024. Available at: <https://www.europol.europa.eu/media-press/newsroom/news/largest-ever-operation-against-botnets-hits-dropper-malware-ecosystem>. Accessed on: 13 Aug. 2024.

[79] SHADOWSERVER. *Shadowserver Endgame*. 2024. Available at: <https://www.shadowserver.org/what-we-do/network-reporting/systembc-historical-bot-infections-special-report/>. Accessed on: 13 Aug. 2024.

[80] FIRST. *FIRST Membership Process*. 2023. Available at: <https://www.first.org/membership/process>. Accessed on: 24 Aug. 2024.

Appendix

The following tables provide specific details of the process described in Section 3 of this work, whose proposal was illustrated in Figure 3. These tables offer information that complements the explanation of the methods and approaches adopted, providing a more detailed view of each step in the procedure. The organization of this data is intended to facilitate the understanding of the phases involved in the process and enable a more in-depth analysis.

Activity	**Inputs**		**Outputs**	
	From	**Description**	**Description**	**To**
Maintain incident database	Ticket System Tool	Creation of an incident database	Incidents entered into a database	Internal
Activities: The organization/institution maintains a malware incident database. The incidents used are previously analyzed and validated by a security analyst and do not contain false positives, only then are they entered into the ticket system.				

Table 19 – Activity: Maintain incident database

Activity	**Inputs**		**Outputs**	
	From	**Description**	**Description**	**To**
Extract incident data	Incident database	Creation of scripts to correctly extract the desired data	Malware incidents	Internal
Activities: Implementation of automated scripts to extract data from security incident repositories, focusing on malware.				

Table 20 – Activity: Extract incident data

Activity	**Inputs**		**Outputs**	
	From	**Description**	**Description**	**To**
Verify incident data	Incident database	Creation of scripts to correctly extract the desired data	Extracted incidents are valid or invalid	Internal
Activities: Evaluate if the extracted data is valid or not. To be valid, the data must be from malware incidents and contain the fields: date and time, institution, and quantity.				

Table 21 – Activity: Verify incident data

Activity	**Inputs**		**Outputs**	
	From	**Description**	**Description**	**To**
Adjust data extraction script	Invalid incident data	The extracted incident data is not valid	Valid incident data	Internal
Activities: Adjust the script to ensure correct extraction of malware incident data, ensuring it contains the fields: date and time, institution, and quantity.				

Table 22 – Activity: Adjust data extraction script

Activity	**Inputs**		**Outputs**	
	From	**Description**	**Description**	**To**
Build and validate neural network model	Malware incident database	Validated malware incidents correctly entered into the database	Neural network using LSTM built and validated	Internal
Activities: Development of a neural network-based model that analyzes time series of malware security incidents. The implementation of LSTM captures complex temporal dynamics and produces accurate predictions. Rigorous tests validate the accuracy and reliability of the model's predictions.				

Table 23 – Activity: Build and validate neural network model

Activity	**Inputs**		**Outputs**	
	From	**Description**	**Description**	**To**
Verify predictions accuracy	Neural network model	Verify if the values being predicted by the model are within an acceptable error margin, less than 50% when analyzing the MAE/mean metric	Prediction accuracy rate	Internal
Activities: Verification of the prediction accuracy rate.				

Table 24 – Activity: Verify predictions accuracy

Activity	**Inputs**		**Outputs**	
	From	**Description**	**Description**	**To**
Optimize model	Values predicted by the neural network model	Implement improvements in the neural network model to reduce the error rate	Optimized neural network model	Internal
Activities: Continuous monitoring of the prediction effectiveness and adjustments to the model as needed to improve accuracy.				

Table 25 – Activity: Optimize model

Activity	**Inputs**		**Outputs**	
	From	**Description**	**Description**	**To**
Save malware incident predictions	Neural network model predictions	Save the predictions made by the constructed neural network model	Predicted values for the next week saved in the database	Internal
Activities: Execution of the model predicts future malware security incidents, additionally, these predicted data are saved in a new database, allowing for future queries and analyses, facilitating access and interpretation of the data.				

Table 26 – Activity: Save malware incident predictions

Activity	**Inputs**		**Outputs**	
	From	**Description**	**Description**	**To**
Extract model-predicted values	Prediction database	Creation of scripts to correctly extract the desired data	Predicted values for the next week	Internal
Activities: Implementation of automated scripts to extract data from the existing prediction repository.				

Table 27 – Activity: Extract model-predicted values

Activity	**Inputs**		**Outputs**	
	From	**Description**	**Description**	**To**
Insert prediction data into visualization tools	Prediction database	Creation of scripts to correctly insert the desired data	Data inserted into visualization tools	Internal
Activities: Create scripts to insert prediction data into information visualization tools.				

Table 28 – Activity: Insert prediction data into visualization tools

Activity	**Inputs**		**Outputs**	
	From	**Description**	**Description**	**To**
Visualize data	Information visualization tool	Use of information visualization tool to create dashboards for anomaly detection in institutions	Detailed prediction data on each client institution	Internal
Activities: Development and/or implementation of a visualization tool that allows the security analyst to inspect and interpret data and predictions in detail. The use of processed information supports proactive decision-making and detailed analyses.				

Table 29 – Activity: Visualize data

Papers Published by the Author

Works published by the author during the program.

Main Submission.

1. Rildo Antonio de Souza, Vitor de Castro Silva, Sylvio Barbon Junior, and Bruno Bogaz Zarpelão **Forecasting Malware Incident Rates in Higher Education Institutions**, Advanced Information Networking and Applications, 04/2024, Springer Cham, 226-237, 978-3-031-57916-5, (Qualis CC 2017, A3)

Complementary publications.

1. Rildo Antonio de Souza **Forecasting Malware Incident Rates in Higher Education Institutions**, Ekoparty Security Conference 2024, Buenos Aires, Argentina, 11/2024. Work presented at the conference.