Czech Technical University in Prague
Faculty of Information Technology
Department of Digital Design

Time Series Analysis of Network Traffic and Detection of Security Threats

by

Ing. Josef Koumar

A Doctoral Study Report submitted to
the Faculty of Information Technology,
Czech Technical University in Prague

Doctoral degree study programme: Informatics

Prague, April 2024

Supervisor:

Doc. Ing. Tomáš Čejka Ph.D.
Department of Digital Design
Faculty of Information Technology
Czech Technical University in Prague
Thákurova 9
160 00 Prague 6
Czech Republic

Abstract

The amount of encrypted traffic is increasing, and the number of readable fields in packet headers is also decreasing. This trend continues to reduce the effectiveness of traditional techniques such as Deep Packet Inspection. Thus, there is an urgent need for innovative methods to ensure cybersecurity. In response, this thesis focuses on threat detection in terms of creating and analyzing time series from network traffic. Our approach complements current research in threat detection in network traffic. Specifically, the thesis introduces two new extensions of IP flows that contain attributes derived from the analysis of packet time series performed directly inside the IP flow exporter. In addition, the thesis presents a new detection method that uses the detection of periodic behaviors. The thesis also explains contributions to the field of anomaly detection, which is the rapidly developing field of cybersecurity research. Finally, thesis propose the author's doctoral thesis's title and topics of interest.

Keywords:
network traffic monitoring, network traffic classification, time-series, time series analysis, Machine Learning

Abstrakt

Množství šifrovaný provozu see stále zvětšuje a počet políček v hlavičkách paketů se taktéž zmenšuje. Tento trend nadále snižuje účinnost tradičních technik jako je Deep Packet Inspection. Tudíž existuje naléhavá potřeba inovativních metod k zajištění kybernetické bezpečnosti. V reakci na to se zaměřujeme na detekci hrozeb z hlediska vytváření a analýzu časových řad ze síťového provozu. Náš přístup doplňuje součastný výzkum v oblasti detekce hrozeb v síťovém provozu. Konkrétně představujeme dva nové rozšířené síťové toky, které obsahují atributy odvozené z analýzy paketových časových řad prováděné přímo uvnitř exportétu síťových toků. Kromě toho představujeme novou detekční metodu, která využívá detekci peridoickébo chování. Tato práce také zasahuje do oblasti detekce anomálií a přispívá k rychle se rozvíjející oblasti výzkumu kybernetické bezpečnosti. Nakonec nastiňujeme budoucí disertační práci autora se zvýrazněním názvu a oblastí zájmu.

Klíčová slova:
monitorování síťového provozu, klasifikace síťového provozu, časové řady, analýza časových řad, strojové učení

Acknowledgement

This work has been partially particularly supported by the Ministry of Interior of the Czech Republic, grant No. VJ02010024: Flow-Based Encrypted Traffic Analysis and also by the Grant Agency of the CTU in Prague, grants No. SGS20/210/OHK3/3T/18 and No. SGS23/207/OHK3/3T/18 funded by the MEYS of the Czech Republic.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Network monitoring plays a crucial role in the overall computer security management. Compared to protections (such as AntiVirus software) deployed on the end devices, the network-based intrusion detection and prevention systems can protect infrastructure against users' sloppiness, policy violations, or (at worst) intentional attacks from the inside.

The Deep Packet Inspection (DPI) can obtain information about activities in a network from packets. However, maintaining network security has become increasingly challenging in recent years due to mass traffic encryption and consequent reduced visibility. Thus, DPI cannot obtain data from packet data parts without decryption using Proxy. Nevertheless, using the proxies for the decryption of traffic is not possible in large infrastructures [7], like ISP networks. Moreover, the encryption of TLS certificates by TLS1.3 [80], deployment of encrypted DNS [39], or Encrypted Client Hello proposal [81] removed the few-remaining information essential for effective threat detection.

Therefore, packet-based monitoring loses a significant part of its benefits, thus, processing packets by packet is mostly unnecessary. Nevertheless, networks are often monitored using the flow-based approach [82], representing the communication between two devices in the form of flows. The flows are created at observation points—network devices aggregating raw network traffic into flow records. The flow records are then transmitted to the flow collector using a flow-export protocol such as the *Internet Protocol Flow Information Export (IPFIX) [21]* or NetFlow Version 9 [20].

The main part of the IP flow is often a vector containing source and destination IP addresses, source and destination transport layer ports, and protocol. Moreover, timestamps (of the first and the last packet in flow) and volumetric information about the flow (number of packets and bytes) are most usually part of the flow records. Aggregation into such flows allows monitoring systems to work even in high-speed ISP networks.

However, Intrusion Detection Systems (IDS) often use DPI which encryption makes unfeasible for monitoring. So, IDS must work with the remaining pieces of information which are mainly volumetric values and timestamps. The combination of these pieces of information can be used to create a time series.

1.1 Motivation

The main motivation for using the time series for network traffic monitoring is that time series are a natural representation of network traffic because each packet/flow has a timestamp and carries some numeric information about the generating process. Furthermore, timestamps are often used as one of the features for the detection of security threats, i.e., a network connection between an infected device and a Command and Control server that can control it. Therefore, using time series analysis on created time series can reveal to be a novel source of information for monitoring systems.

The statistical view at time series can be used for multiple purposes in network monitoring, for example, network traffic classification or anomaly detection in network traffic. However, the deployability of current time series analysis techniques in real network environments is not sufficiently addressed in the state of the art. Moreover, the larger the target network, the more complex the problem becomes, especially since the deployment of time series analysis methods in high-speed ISP networks is a crucial challenge.

1.2 Problem Statement

This report aims to find suitable algorithms and mathematical methods that will be used to analyze high-speed network traffic, including encrypted communication, using time series analysis. The topic uses the fact that the regularity and temporal characteristics of communication significantly depend on the types of applications and their use. Network communication often exhibits the necessary periodic behavior observable at the network layer with the aim of regular communication, status updates, etc. The detection of security threats is of high importance for the defense of the network infrastructure. The analysis of time series from network communication is a non-trivial task for which there is currently no reliable universal solution. Moreover, in the network traffic environment, there are usually large volumes of data that must be processed in a short time (due to early detection), which complicates the solution. The basis will be research in the field of the possibilities of using statistical methods, probabilistic models, and possibly the use of artificial intelligence algorithms.

1.3 Structure of the Report

The report is organized into the following chapters. Chapter 2 *Background and State-of-the-Art* introduces time series analysis in the network traffic monitoring domain and summarizes existing works. Chapter 3 *Overview of Thesis Approach* presents the thesis approach for network traffic classification using time series analysis. Chapter 4 *Preliminary Results* provides a description of achieved results. Finally, Chapter 5 *Conclusions* describes the intended topics of the doctoral thesis and concludes the report.

Chapter 2

Background and State-of-the-Art

This chapter provides a theoretical background to network traffic monitoring, time series analysis, and also an overview of the related published works in the context of using time series analysis for network traffic monitoring and detection of security threats. Time series analysis of network traffic is the primary scope of the future dissertation thesis. The most related papers and topics are grouped into several sections to cover the area systematically.

2.1 Theoretical Background for the Network Traffic Monitoring

Network traffic monitoring plays a crucial role in overall security. The firewall may protect the infrastructure from attacks from outside the company, and antivirus software may detect threats on end devices. However, the firewall and antivirus software cannot detect all types of threats. Therefore, network traffic monitoring complements these protections and enhances their functionality.

The system that uses network traffic monitoring for the detection of security threats is called an Intrusion Detection System (IDS) and it requires additional infrastructure to build. First, the network traffic usually must be mirrored, for example, on the router using a mirroring port or using a device called network TAP (Test Access Point). The traffic is mirrored to another device called a monitoring probe. In small networks, the monitoring probe can be part of the IDS system on the same server. However, this is usually not possible in large infrastructures because the network usually has a redundant connection to the internet. Thus, multiple monitoring probes are deployed in such a network, and probes send the traffic (raw or aggregated into flows) into a server called a collector, which is part of the monitoring system. An example of such monitoring infrastructure is shown in Figure 2.1.

As was briefly explained in the introduction, network traffic uses encryption on most of its content. So, maintaining network security has become increasingly challenging in recent years due to mass traffic encryption and consequent reduced visibility. The encryption of TLS certificates by TLS1.3 [80], deployment of encrypted DNS [39], or Encrypted Client

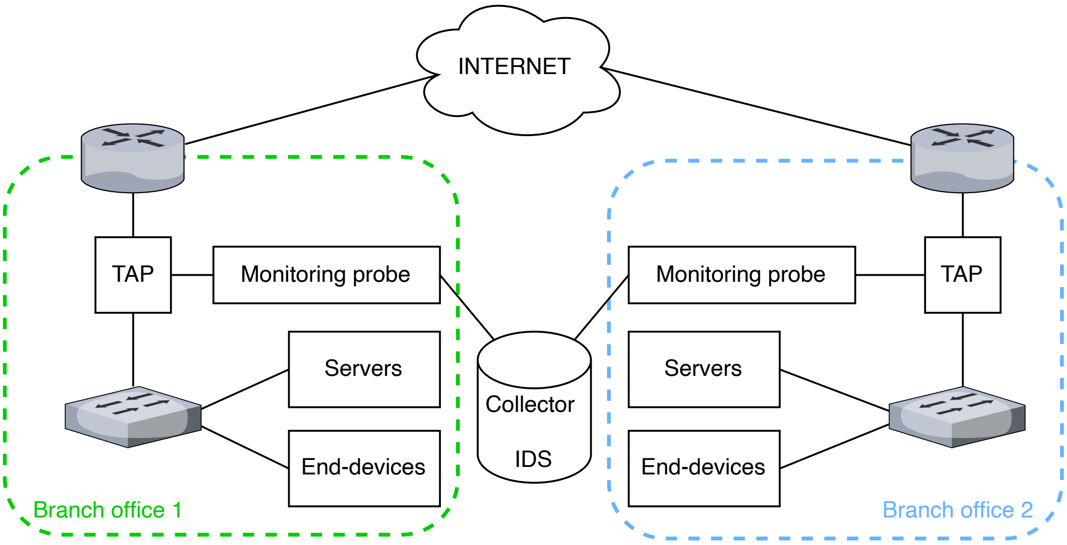

Figure 2.1: Example of the monitoring infrastructure with multiple monitoring probes

Hello proposal [81] removed the few-remaining information essential for effective threat detection. Therefore, if possible, the security managers are forced to deploy intermediate proxies to decrypt the traffic [71] and inspect it via Deep Packet Inspection (DPI) tools such as Suricata[1]. In such cases, the deployment of the intermediate proxy is much more intrusive than sending domain names and certificates in plaintext. The DPI combined with the proxy is an efficient solution for mid-sized restricted networks. However, it does not scale to large provider-based networks, where threat detection is also desired [7].

2.1.1 Flow-based monitoring

Internet Service Providers (ISP) need to perform network monitoring and cybersecurity threat detection to force internal policies, infrastructure protection, and lines overloading prevention to maintain service [7]. The single intrusion detector deployed at the ISP level can protect many users (even in order of millions) and prevent or minimise the impact of attacks, such as DDoS. In ISP deployment, the use of the intermediate proxy is unthinkable. It would outrage the consumers due to absolute payload availability; moreover, it is not feasible to process such a large amount of traffic transmitted over multiple 100 Gbps backbone lines with DPI. Therefore, large-scale infrastructures are often monitored using the flow-based approach, representing the communication between two devices in the form of flows.

Flow-based network monitoring is essential for security maintenance in large network infrastructures. The flow-based intrusion detection systems deploy a variety of classifiers and detectors of malicious communications. Naturally, the design of these detectors strongly

[1] `https://suricata.io`

depends on the information available in the flow. The flows can contain almost any data that can be extracted from the communication [40]. Modern flow export protocols such as IPFIX or NetFlow Version 9 support templating mechanisms so that the users can define their data structures transferred inside flow records. Thus, the flows are often extended for various information helpful in intrusion detection.

2.1.2 Detection of security threats

Network traffic monitoring plays a critical role in the detection and analysis of security threats. Security systems can identify patterns or anomalies indicative of malicious activities by examining the packets or IP flows in the network. Among various methods, monitoring IP flows stands out for its efficiency and effectiveness in threat detection.

By analyzing IP flows, security analysts can discern not only the volume of traffic between two points but also the nature of the traffic. This is pivotal in detecting a wide range of security threats, from distributed denial-of-service (DDoS) attacks, where a target is overwhelmed with traffic from multiple sources, to more sophisticated intrusion attempts that may seek to exploit specific vulnerabilities within a network.

The analysis of IP flows facilitates the identification of unusual traffic patterns, such as spikes in traffic to particular ports or destinations that could indicate a breach attempt or the presence of malware. Additionally, IP flow data can be used to construct baselines of normal network behavior, enabling the detection of deviations that might signal an ongoing attack. For instance, an unexpected request to a high-value server from an unfamiliar external IP could be flagged for further investigation.

However, while IP flow monitoring is a powerful tool in the cybersecurity arsenal, it does have limitations. Encryption, for example, can obscure the contents of IP packets, making it challenging to determine whether a flow is benign or malicious based purely on header data. Thus, IP flow analysis is often complemented by other forms of network and security monitoring, such as behavioral analytics, to provide a more comprehensive view of network security. Moreover, most of the detection is today done by using machine learning. Machine learning or deep learning can capture the statistical properties of malware communication despite the traffic being encrypted.

In conclusion, the monitoring of IP flows is a fundamental aspect of modern network security strategies. By providing detailed insights into traffic patterns and behaviors, IP flow analysis helps security teams detect and respond to potential threats more swiftly and effectively. Despite its challenges, when integrated with other security measures, it forms an essential layer of defense in the protection of digital assets and infrastructures.

2.1.3 Conclusion of network traffic monitoring State-of-the-Art

Network traffic monitoring is a well-established research area with speed changes due to trends in network attacks. Furthermore, due to the general increase of traffic on the internet and also due to the increasing speed which can network infrastructures carry, increased requirements for monitoring infrastructure. Thus, intrusion detection systems must work faster. Moreover, IDS must be also more precise because 1% of false positives in a slow network results in several mistakes per minute, however, on a fast network this false positive rate results in thousands of mistakes per minute, which is much more challenging number for human security analysts.

Therefore, researchers must develop more precise methods or combine existing methods into one that also can handle high-speed network traffic. Overall precision can also help in looking at larger time intervals, which are harder to handle due to a lot of data to store and process. Therefore, this thesis presents an idea that time series can be used to address these issues and can be one of the components of such a system.

2.2 Theoretical Background for Time Series Analysis

First, the definition of a time series (TS) is required but primarily, another term has to be defined: Stochastic Process. [103] provides a comprehensive definition:

Definition 2.2.1 (Stochastic process) *A stochastic process is a family of time-indexed random variables $Z(\omega, t)$, where ω belongs to a sample space and t belongs to an index set.*

A time series is a realization of a certain stochastic process. A formal distinct definition for TS is as follows:

Definition 2.2.2 (Time series) *A time series is a sequence of observations taken by continuous measurements over time. Generally, the observations are picked up in equidistant time intervals:*

$$\mathcal{T} = \left(t_0^d, t_1^d, \ldots t_t^d\right), d \in \mathbb{N}_+, t \in \mathbb{N} \tag{2.1}$$

where d defines the dimension of the time series.

So a TS can be a sequence of observations from some monitored process. If $d = 1$, then TS is called univariate, another TS is called multivariate and each dimension is usually denoted as the time series variable. The dimensions of multivariate TS are often called TS metrics. Thus, the univariate TS is a time series with one TS metric. If $t \in \mathbb{N}$ then TS is discrete, other for example $t \in \mathbb{R}$ TS is continuous. Also, TS are usually split by spaces between timestamps $t_0, t_1, \ldots, t_t$ into evenly sampled TS and unevenly spaced TS. Both of these types of time series are deeply described in Sections 2.2.1 and 2.2.2.

Furthermore, time series can be interpreted as a combination of several components:

$$Y_t = T_t + S_t + \epsilon_t \tag{2.2}$$

The Y_t is the value of the TS metric at timestamp t. The T represents the trend component, S represents the seasonal component, and ϵ represents the random component. Moreover, these TS components are part of important properties which are deeply defined below.

Definition 2.2.3 (Trend) *A time series has a trend, if its mean, μ, is not constant, but increases or decreases over time. A trend can be linear or non-linear. The time series in Figure 2.2 has a positive trend from 2005 until 2008, and a negative trend afterward.*

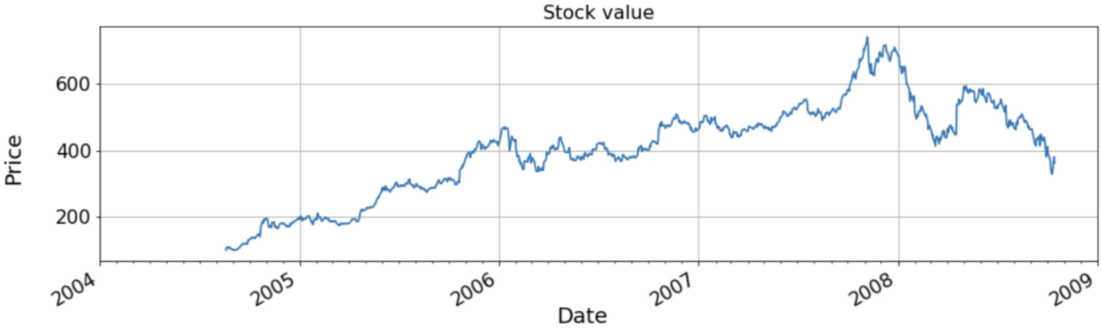

Figure 2.2: Sample TS showing the prices of a stock over five years [16]

Definition 2.2.4 (Seasonality) *Seasonality is the periodic recurrence of fluctuations. The time series is called seasonal because seasonal factors like time of the year or day of the week, or other similarities are influencing it.*

Thus, it always has a fixed period of time that is limited to a year. Figure 2.3 shows a seasonal time series. It is the monthly home sales index for 20 major US cities between the years 2000 and 2019.

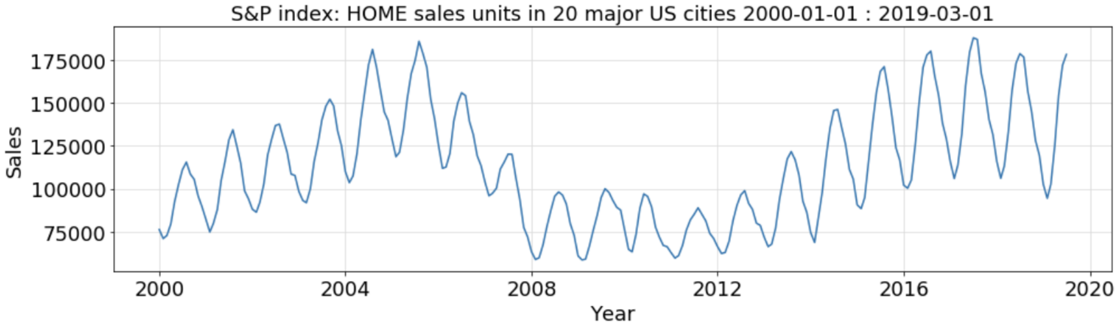

Figure 2.3: Sample TS with strong seasonal component

Definition 2.2.5 (Cycles) *A cyclic time series is influenced by time factors where the period is not fixed and the duration is above a year, e.g., a decade.*

The time series in Figure 2.3 also has an approximate 12-year cycle.

Definition 2.2.6 (Level) *The time series level is equal to the mean of the series. If a time series has a trend, then it is often said that the level is changing.*

Intuitively, a stationary time series is a time series having the same characteristics over every time interval. The formal definition [45] says, that:

Definition 2.2.7 (Stationarity) *X_t is a stationary time series, if $\forall s \in \mathbb{R}$ the distribution of $(x_t, \ldots, x_{t+s})$ is equal.*

The above definition implies that a stationary time series $x_1, \ldots, x_T$ will have the following characteristics:

1. **Constant mean**, thus no trend exists in the time series.

2. The time series has a **constant variance**.

3. There is a **constant autocorrelation** over time.

4. The time series has **no seasonality**, i.e., no periodic fluctuations.

The stationarity is an important property because some Time Series Analysis (TSA) methods assume stationary time series as input [78]. An example of such a TSA method is the AR (Auto-Regressive) model for anomaly detection.

2.2.1 Evenly sampled time series

The evenly sampled time series are so dominant in the literature that when time series is written it usually means evenly sampled time series [78]. Furthermore, evenly sampled time series can be also referred as uniformly spaced or regularly sampled in literature. Naturally, uniformly sampled TS occur in many fields and usually are preferred because it is easier to handle them by existing methods and tools. Examples of such fields are finance, retail, economics, historical data, geological, weather, and medicine. The formal definition of evenly sampled TS is presented below.

Definition 2.2.8 (Evenly sampled time series) *A series of values of a quantity obtained at successive times with equal intervals between them. Therefore, it is true that:*

$$\forall x_n \in \mathbb{R}, \forall t_n \in \mathbb{R}, \forall n \in \mathbb{N}, t_n - t_{n-1} = t_{n+1} - t_n \tag{2.3}$$

where n is index, x_n is the value (or vector), and t_n is timestamp.

The substitution and division can be applied on time series timestamps as follows:

$$\forall t_n \in \mathbb{R}, \forall n \in \mathbb{N}, \frac{t_n - t_0}{t_1 - t_0} \tag{2.4}$$

The Equation (2.4) will ensure that for new timestamps t is true that $t \in \mathbb{N}$. Moreover, the new timestamps now only represent the sequence numbers of data points x_n. Therefore, the evenly sampled time series can be interpreted only as a sequence of observation x_n.

2.2.2 Unevenly spaced time series

The unevenly spaced time series naturally occur in many industrial and scientific domains like astronomy, natural disasters (earthquakes, floods, etc.), medicine, economics, signal processing, IoT data, and so on. Furthermore in literature, they can be found as an unequally spaced or irregularly sampled time series [13]. The formal definition of unevenly spaced time series is presented below.

Definition 2.2.9 (Unevenly spaced time series) *A series of values of a quantity obtained at successive times, where at least one interval between them differs from others. Therefore, it is true that:*

$$\forall x_n \in \mathbb{R}, \forall t_n \in \mathbb{R}, \exists n \in \mathbb{N}, t_n - t_{n-1} \neq t_{n+1} - t_n \tag{2.5}$$

where n is index, x_n is the value (or vector), and t_n is timestamp.

It is obvious that the Equation (2.4) cannot be true for all $n \in \mathbb{N}$. Therefore, the methods and tools for the unevenly spaced time series must work with both values/vectors and timestamps simultaneously.

2.2.3 Time Series Analysis

Time series analysis is a crucial statistical tool used across various scientific fields to model the dynamics of sequential data points collected over time. This analysis is fundamental in disciplines such as economics, engineering, environmental science, and more, offering insights into underlying patterns, forecasting future trends, and understanding temporal relationships within data.

Key topics in time series analysis include descriptive techniques, probability models, forecasting, spectral analysis, bivariate processes, linear systems, state-space models, Kalman filter, non-linear models, and multivariate time series modeling [19]. These methods are employed to analyze time series data in both the time and frequency domains, with a rich set of references for further reading [15].

In research, time series analysis is particularly beneficial in studying climate variability and change, as it allows for extracting valuable information that simpler methods may not reveal [33]. Techniques such as the classical time series decomposition, exponential smoothing method, stationary and nonstationary series analysis, the Box-Jenkins method, and their applications demonstrate the versatility and depth of time series analysis in addressing real-world problems.

Moreover, modern techniques of multiple time series analysis aid in assessing relationships between series, highlighting the role of time series data, including stock prices, weather conditions, and GDP in analysis and forecasting [36]. The development and use of ARIMA, structural, and stochastic volatility models are crucial for prediction and forecasting in finance, insurance, and other fields [36].

From a technological perspective, recent literature analysis between 2017 and 2021 indicates a significant focus on the classification and prediction of time series data, demonstrating the growing importance of time series analysis in developing advanced analytical tools and techniques [99].

Time series analysis, with its emphasis on modeling dependencies across time and its wide range of applications, continues to evolve, incorporating advanced statistical and machine learning techniques to address complex challenges in scientific research and practical applications.

The time series analysis [78] can show how variables change over time, which is crucial because it shows dependencies between data. Most approaches are used to help to understand the underlying causes of trends or patterns. Also, it is possible to use time series forecasting to predict the future. Usually, TSA requires a large number of data points in TS to ensure consistency and reliability. If TS contains a small number of data points, then the detected trend or periodicity can be false and caused by noise.

There are several types of time series analysis, based on what is the goal of the analysis. The main goal is forecasting which is used for predicting future data based on historical trends. There are many approaches to time series forecasting [107, 50, 63, 5, 48]. Similar approach which sometimes uses forecasting is anomaly detection [12, 51, 53, 84]. Detecting anomalies in time series. It uses historical data to train the model, if a new data point doesn't fit into the model, it is reported as an anomaly.

Moreover, time series are used also for the classification [95, 1, 46, 106, 110]. The most used technique for classification in the time series domain is time series clustering which involves the identification and categorization of data points within a dataset. Through classification, data is organized into distinct groups based on shared characteristics, facilitating further analysis and understanding of the data's structure and composition. For example, Susto et al. [95] introduce the time series classification problem, describe the features extraction procedure with mostly used features for classification, and also present a list of classification approaches for time series classification with complexity and comparison. The presented methods are feature-based or distance-based.

The next method of time series analysis is called curve fitting. By plotting data points along a curve, curve fitting examines the relationships between variables within the dataset. This process is crucial for understanding how different variables interact with each other and can help in predicting future trends based on these relationships.

The descriptive analysis technique focuses on identifying patterns within time series data, such as trends, cycles, and seasonal variations. Descriptive analysis is foundational in time series analysis as it allows for the summarization of key features of the data, providing insights into the overall behavior of the dataset over time [97, 17, 108, 65, 92, 8].

Beyond merely describing data, explanative analysis seeks to uncover the underlying causes and effects within the dataset. By understanding the relationships and interactions within the data, this method helps in formulating theories about why certain patterns or trends occur. Furthermore, the exploratory analysis emphasizes the visualization and summarization of the main characteristics of the time series data. Exploratory analysis is often the first step in data analysis, offering a preliminary overview that guides further, more detailed investigations.

Moreover, identifying recurring patterns or cycles within time series data is the focus of periodicity detection. This method is vital for understanding the regularity of patterns and predicting future occurrences based on these identified cycles [25, 75, 79, 93, 104, 38].

A typically used method to detect periodicity is a periodogram defined by Arthur Schuster in [88]. The periodogram creates a spectral density of a signal and it is used to identify important frequencies of a time series. There are many versions derived from the original periodogram, e.g., Bartlett's procedure [11], Welch method [10], Laplace periodogram [57] and Lomb-Scargle periodogram [100].

Another typical method of detecting periodicity is the autocorrelation function, which is based on checking a correlation of a time series with a lagged version of itself. The origin of correlation was explained by Pearson in the article [72].

There are also methods of detecting periodic behavior that are rather engineering than mathematical, for example, an apriori [4]. This technique of frequent pattern mining use two steps called "join" and "prune" to reduce a searching space and these steps are iterated in loops. The apriori was also improved into periodicity mining [37].

Often one step of the time series analysis is to transform data to highlight features or remove unwanted features. This is often done by taking the log transform or a linear transform. A transformed time series can be often easier to analyze or contain features not apparent in the original time series.

Exponential smoothing methods are one of the oldest, simplest and at the same time very effective ways of working with time series. Using them, it is possible to easily interpolate and extrapolate (predict) the development of the series, which may have different components, including trend or seasonality. A simple (or single) exponential smoothing (SES) [70], also Holt's linear method, is a basic method that assumes the absence of trend and seasonality. A double exponential smoothing (DES, or double ES) [14] is an extension of SES to time series with a trend. There are more DES methods, for example, the Holt method with a linear trend and the method with a damped trend.

2.3 Time series analysis in network traffic

Time series analysis plays a pivotal role in the domain of network traffic monitoring, offering comprehensive insights into traffic patterns, predicting future network behaviors, and ensuring the smooth functioning of network infrastructures. Therefore, most approaches use time series for forecasting or anomaly detection. However, the time series are used for many other purposes, like traffic classification or security threats detection.

This section is divided into several subsections. First, the network traffic forecasting problem is introduced, and then it continues with the closely related topic of anomaly detection in network traffic. Furthermore, the network traffic classification and usage of time series and their analysis in this domain is introduced.

2.3.1 Network traffic prediction

The field of network traffic forecasting is rapidly evolving, with several notable trends shaping its future. Firstly, the integration of deep learning models, such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), has become increasingly prevalent; these models excel in capturing complex spatial and temporal patterns in traffic data, leading to more accurate and dynamic predictions [66, 77]. Secondly, there is a growing emphasis on hybrid models that combine different machine learning techniques to leverage their unique strengths, such as the fusion of CNNs for spatial feature extraction with Long Short-Term Memory (LSTM) networks for capturing temporal dependencies [66]. Thirdly, the application of graph neural networks (GNNs) is emerging as a powerful tool for modeling the network's topology, enabling predictions that account for the interconnected nature of network nodes and paths.

Another significant trend is the focus on real-time traffic forecasting, driven by the need for immediate decision-making in dynamic network environments. This requires the development of models that can efficiently process streaming data and adapt to changes in traffic patterns. Additionally, there is an increasing reliance on big data analytics and the incorporation of diverse data sources, including Internet of Things (IoT) devices and social media, to enrich traffic predictions with contextual information [41].

Moreover, the use of statistical methods and traditional time series analysis continues to play a crucial role in network traffic forecasting, offering robust mathematical frameworks for understanding traffic dynamics [42]

Lastly, the explainability and interpretability of models are gaining attention. As network traffic forecasting increasingly relies on complex machine learning models, the ability to interpret these models and understand the rationale behind their predictions becomes essential. This need for transparency helps to build trust in the models' outcomes and facilitates their practical implementation in network management [109].

These trends underscore the movement towards sophisticated, accurate, and adaptable traffic forecasting methods, supported by advances in machine learning, statistical analysis, and data analytics. As networks grow in complexity and scale, these developments are crucial for ensuring efficient and reliable network operations.

2.3.2 Anomaly detection in network traffic

As digital technology and internet usage advance, the frequency and complexity of cyberattacks have also increased, targeting network vulnerabilities to access sensitive information or disrupt legitimate operations. Supervised detection of threats can detect only some parts of threats that can be detected and are known. Therefore, unsupervised detection methods must be implemented to detect unknown or zero-day threats. These unsupervised detection methods implement some anomaly detection techniques. Anomaly detection methods usually model a baseline which is used for comparison with real observed traffic. Thus, time series forecasting is the main anomaly detection method used today.

Anomaly detection in network traffic can be done by large possible methods. For example, Kumari et al. [54] propose a solution employing unsupervised learning techniques based on k-means clustering for anomaly detection in network traffic that could indicate a potential intrusion. The results show that k-means clustering, especially when coupled with Spark's real-time data processing capabilities, can serve as a robust tool for anomaly detection in network traffic. Also, Dornel et al. [23] describes that exponential smoothing, AR, and ARIMA models are utilized to estimate and predict both incoming and outgoing traffic, facilitating real-time anomaly detection and proactive network management for enhanced decision-making in network operations.

Moreover, Vikram et al. [101] focus on the application of the Isolation Forest algorithm for anomaly detection within network traffic. The Isolation Forest algorithm, known for its tree-based structure that isolates outliers effectively, is leveraged to identify potential attacks by examining the path length in the trees — shorter paths typically indicate anomalies. Results show that Isolation Forest is a viable solution for improving the accuracy and efficiency of network intrusion detection systems.

Nevertheless, Simmross-Wattenberg et al. [90] presents a novel method for detecting anomalies in network traffic by leveraging statistical inference and α-stable modeling. Traditional models like Poisson and Gaussian distributions do not adequately capture the complex behaviors of network traffic, such as its inherent burstiness and heavy-tailed distributions. Thus, Simmross-Wattenberg et al. [90] propose using α-stable distributions, which can model the high variability and asymmetry found in real network traffic, making them a promising tool for anomaly detection. Moreover, the approach finds statistical evidence supporting the use of α-stable distributions as a model for network traffic marginals and demonstrates the method's effectiveness in identifying potential intrusions by comparing it with other models and approaches.

Furthermore, Ma et al. [62] proposes a new anomaly detection model for network traffic using machine learning techniques. The core of the model, referred to as SVM-L, leverages raw URLs as natural language, transforming them into mathematical vectors through statistical laws and natural language processing techniques. These vectors then serve as training data for a traffic classifier based on the kernel Support Vector Machine (SVM). A notable innovation in this model is the optimization model proposed for hyper-parameter adjustment of the classifier, specifically tailored for the dual formulation of kernel SVM and Linear Discriminant Analysis (LDA). This approach simplifies the problem to a one-dimensional optimization easily solved by the golden section method. The authors report that numerical tests demonstrate the model's superior accuracy over 99% on all tested datasets, outperforming state-of-the-art techniques in standard evaluation measures.

And finally, for example, Hwang et al. [44] proposes an advanced method for detecting anomalies in network traffic. The proposed model, named D-PACK, integrates a Convolutional Neural Network (CNN) with an unsupervised deep learning model, such as an Autoencoder, to automatically profile traffic patterns and identify anomalies early by inspecting just the initial bytes of the first few packets in each flow.

Therefore, there is a huge number of methods used for anomaly detection in network traffic. However, many of them are tested only on offline data and do not address deployability in real-world scenarios.

2.3.3 Classification of network traffic

The classification of network traffic to gain a view of encrypted traffic is a well-established domain. The purpose of the research is, for example, the detection of security threats in IDS systems, the detection of VPN or TOR traffic, and also service classification in TLS or QUIC traffic.

The application of time series analysis in threat detection on network traffic is increasingly crucial in identifying and mitigating network attacks. Researchers have developed various methodologies using time series analysis to enhance the detection of these threats in high volumes of network traffic. In recent years, machine learning and deep learning techniques have a high popularity for the classification of network traffic for security threat detection.

One of the most used representations of network traffic is the Sequence of Packet Lengths and Times (SPLT). The SPLT is a critical feature set used in the analysis and classification of network traffic. At its core, SPLT represents a sequential log of the sizes (lengths) of packets and the timing information (inter-arrival times) between these packets as they are transmitted over a network in one IP flow. Therefore, the unevenly spaced time series of packets transferred at particular times is transferred into a sequence of lengths and interarrival times creating the evenly sampled multivariate time series with two time series metrics. The SPLT provides a detailed fingerprint of network activity, capturing the unique patterns that different applications and services imprint on network traffic.

This approach encompasses a variety of techniques and methodologies, tailored to address the inherent challenges associated with encrypted network traffic classification and the broader scope of network traffic management. Moreover, the SPLT always captures only the first n packets in IP flow data, therefore, it can be easily exported by IP flow exporters which makes this approach highly important in this domain.

The exploration of the SPLT for security threat detection within network systems has been detailed in various scholarly works, each presenting unique methodologies and findings. The overarching goal in these studies is to leverage SPLT data to identify, categorize, and mitigate potential security threats, ranging from malware dissemination to distributed denial-of-service (DDoS) attacks.

Anderson et al. [6] use of the SPLT for identifying threats within encrypted network traffic. SPLT, along with byte distribution, forms part of the observable metadata used to develop the supervised machine learning models. This approach is aimed at capturing the unique patterns that different applications and services imprint on network traffic, which are essential for distinguishing between malicious and benign flows without decrypting the traffic.

A similar approach was presented by Vu et al. [102] which extracts packet features that strongly represent encrypted packets for deep-learning network traffic classification and then arranges receiving packet samples in a time-series order. Furthermore, they leverage the Long Short-Term Memory (LSTM) network to extract the time dependence of time series packets automatically. Also, Safeghzadeh et al. [83] use first m packets of flow as input of neural network but as two uniformly sampled time series, one for packet size and one for inter-arrival times between packets. Moreover, Montazerishatoori et al. [67] use uniformly sampled multivariate TS created from flow records with one variable that describes time differences between the flows. This approach is used to identify tunneling activities that utilize DNS communications over HTTPS by presenting a two-layered approach to detect and characterize DoH traffic using time-series classifiers.

Moreover, Aceto et al. [3] utilizes a deep learning architecture that automatically extracts features from the SPLT, exploiting its multimodal nature—meaning the framework can process traffic data from multiple sources or formats simultaneously. The approach was used for the VPN traffic classification.

Furthermore, Luxemburk at al. [59] presents a comprehensive approach to the classification of encrypted traffic, specifically focusing on TLS services. The neural network architecture described integrates 1D convolutional and linear layers, processing both SPLT and flow statistics. Moreover, a similar approach was used by Luxemburk at al. [61] also for the classification of QUIC services.

However, the SPLT and similar packet time series representation of IP flow which serves as input into a neural network is not the only one methodology of using time series in the network traffic classification domain. For example, Vu et al. [102] developed a novel time series feature extraction technique to address the encrypted traffic classification problem. First, a feature engineering technique is applied to network traffic to extract significant attributes of the encrypted network traffic. These significant attributes include bytes from the application layer or pieces of information from network and transport headers which are organized into time series. Next, the time series serves as input into LSTM networks to capture the temporal dependencies in the data. This approach aims to efficiently represent the behavior of encrypted network traffic, allowing for accurate classification.

Moreover, Bai et al. [9] aims to enhance IoT device management by enabling automatic identification of device types. In this approach, the network traffic flows are segmented into fixed time intervals from which are extracted features, focusing on aspects like packet numbers, packet lengths, network protocols, and the direction of traffic. These features are designed to capture the behavioral patterns of IoT devices based on their network activity. Leveraging a Long Short-Term Memory (LSTM) — Convolutional Neural Network (CNN) cascade model, the extracted features are used to classify the IoT devices into predefined categories.

Furthermore, Abdullah et al. [2] presents a novel approach to classify network traffic and identify anomalies by leveraging the principles of time series and fuzzy logic. The network traffic is aggregated into time series which are data input into the approach. The approach employs a fuzzy inference system to classify network traffic into normal or attack categories based on the deviation from the modeled normal behavior. A significant feature of the approach is its evolving capability, where the system's knowledge base can adapt and grow based on new data, thus improving its accuracy over time.

Furthermore, Eslahi et al. [26] aims to detect periodicity in HTTP traffic filtered to GET and POST methods. A simple decision tree assigns five tags (classes) based on the type of observed periodicity. Based on these tags, the authors decide if HTTP traffic is generated by a botnet or not. The paper [76] purposes a novel detection model named detection by mining regional periodicity (DMRP), which is used to detection of P2P botnets.

Moreover, Haffey et al. [35] shows that periodic traffic analysis is effective for detecting P2P, gaming, cloud, scanning, and botnet traffic IP flows. For detection periodicity, they are using SQL-based implementation of the periodicity detection method proposed by Hubballi and Goyal [43]. Also, Schatzmann et al. [86] uses periodicity as one of the three features to identify webmail traffic.

Chapter 3

Overview of Thesis Approach

The thesis aims to design methods and tools for security threat detection in network traffic using time series analysis. The thesis discovered the missing opportunities in the current research using time series analysis in this domain. Most of the approaches that focus on network traffic classification based on time series analysis, use the Sequence of Packets, Lengths, and Times as input into deep learning. However, time series analysis can be designed for feature engineering as well. Thus, the thesis focuses in first half on classification based on features obtained by time series analysis.

Furthermore, the use of time series analysis for network traffic forecasting and anomaly detection is a well-established area with many methods that achieve great precision. However, most approaches were tested on datasets not created for evaluating anomaly detection methods. Furthermore, the datasets are also laboratory-created. Thus, the research community does not know how most of the approaches work in the real world. Moreover, most of the approaches do not refer to deployment into real-world scenarios. Therefore, the thesis focuses the second half on evaluating anomaly detection in real-world scenarios and creating real-world big-data datasets.

3.1 Time Series from Network Traffic

Time series analysis of network traffic can reveal useful information about the status and events in the network. However, suitable traffic representations and models are required for such detection. Therefore, the thesis focuses on the creation of a novel time series representation of network traffic and evaluates the properties of the novel one with classical ones.

3.1.1 Creating Time Series from Network Traffic

Creating a network traffic time series is more complex than it may seem at first glance. It is always necessary to divide the traffic so that only the examined process is present in the time series. Therefore, first, it must be decided for what time series will be used. It is a simple task to analyze the activity of a single device or network.

Nevertheless, the task becomes problematic when is necessary to directly analyze one specific process or application running on a particular device. An example can be the browser's communication with any web page. If it is possible to have full access to the device on which the browser is running, then the time series can be created easily from logs of processes. However, the entire network's traffic is monitored with the assumption of no access to the devices, so the browser traffic must be identified in the traffic of one or more networks.

The thesis proposes an approach that deals with it. The approach uses IP addresses to identify specific devices. A particular registered or well-known protocol can also be used, using ports of the transport layer. These features divide the analyzed traffic into network dependencies, which were designed particularly in this thesis for this purpose. Network dependency is a long-term communication of pairs of devices, where one device provides service to another. So, by using network dependencies, is possible to merge multiple packets or flows (i.e., connections) into a single time series.

Once the traffic is divided, creating a time series is straightforward. If it is necessary to create unevenly spaced time series, then each packet or flow is put into its position in the time series based on their time information. If it is necessary to create an evenly spaced time series, then some aggregation technique must be used to aggregate packets or flows into a vector of attributes for each time window.

3.1.2 Evenly sampled Time Series

In the field of network traffic forecasting and anomaly detection, it is a well-established methodology for using evenly spaced time series. However, this thesis faces several challenges for this type of time series.

First, the aggregation process requires proposing suitable attributes that will be generated from the raw traffic, i.e., packets or IP flows. Nevertheless, the choice of attributes is highly dependent on a task. However, the results of the approach can also be highly influenced by the attributes of the choice; for example, if the anomaly detection system will be working only with an aggregated number of packets, then it cannot detect when the end device starts to communicate with more end devices in the same network. Thus, in this example, the anomaly detection system will miss the possible scan from the end device.

Secondly, the aggregation into evenly-spaced windows then, with high probability, occurs a window where non-traffic will be transferred. Therefore, the time series will contain zero values. However, mathematic methods can be influenced by these zero intervals and their precision will drop.

Moreover, suppose an approach will aggregate network traffic in time intervals as it is used in most published papers. In such a case, it gets a time series with high dependence on the aggregation interval. For example, the evenly spaced time series was created with aggregation interval 60 seconds in Fig. 3.1. However, we can see that the explicit cyclic behavior of IP flows in unevenly spaced time series is noised by aggregation. Thus, the key behavior was lost if the task was classified.

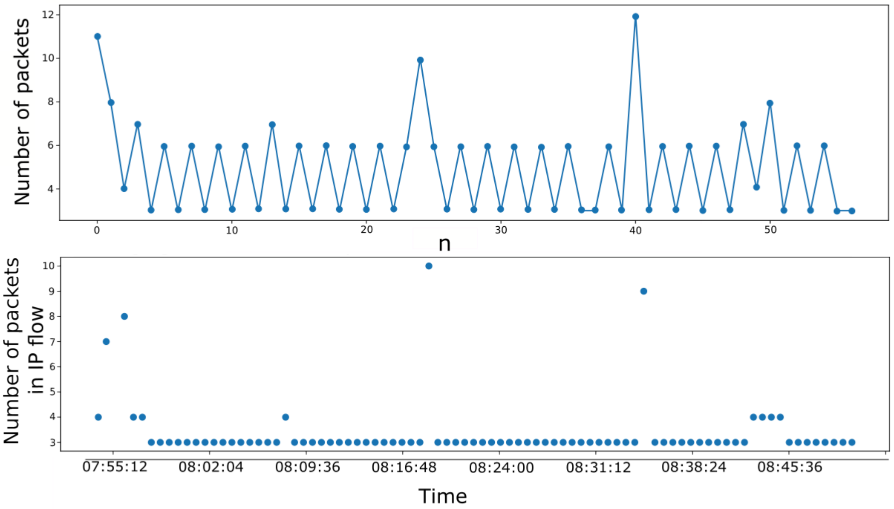

Figure 3.1: Comparison of unevenly spaced time series created by aggregation (top graph) and raw unevenly spaced time series (bottom graph)

3.1.3 Unevenly Spaced Time Series

Contrary to the currently used approaches, this thesis proposes Unevenly Spaced Time Series as a feasible representation of network traffic in time with several benefits for analysis. The thesis defines multiple types of Unevenly Spaced Time Series from network traffic, designated by what represents one data point and what data points have in common.

3.1.3.1 Packet Time Series

A Packet Time Series from network traffic is a time series where a data point represents a network packet. Furthermore, Packet Time Series is a univariate time series with a variable number of bytes in the network packet. The time information $t_i \in \mathbb{R}$ of i-th data point is defined by the transmission time of i-th packet. So Packet Time Series are inevitably Unevenly Spaced Time Series.

3.1.3.2 Flow Time Series

A flow represents aggregated information from a sequence of packets with the same attributes in the packet headers. The Flow Time Series are multi-dimensional to cover all valuable information, such as the number of packets and bytes. Therefore, such Flow Time Series are multivariate Unevenly Spaced Time Series. Another important fact is that the flow record contains two timestamps, namely the time of the first and the last packet of the flow, so the Flow Time Series has two time axes. These two time axes can generate additional Flow Time Series variables, which can be well applied in analysis and possible classification. The first axis is a variable duration of the flow. The second one is a variable time difference representing the gaps between flows.

3.1.3.3 Single Flow Time Series

Since PTS may contain packets of any connection together, creating separate TS is useful. Such TS that contains a packet sequence of just a single flow is called a Single Flow Time Series. Because a Single Flow Time Series is a special case of a Packet Time Series of one flow only, it cannot contain network traffic of more than one process, and the only noise that can occur is packet retransmission. Therefore, the noise in the Single Flow Time Series is minimal.

However, the Single Flow Time Series created by the sizes of packets and their timestamps do not have evenly spaced timestamps between the datapoints. That means a time series of observations $\{X_n\} = (x_1, x_2, \ldots, x_n)$ taken at times $\{T_n\} = (t_1, t_2, \ldots, t_n)$ does not have constant $\delta_j = t_{j+1} - t_j, \forall j \in \{1, \ldots, n-1\}$. Thus, analysis requires methods designed for unevenly spaced time series.

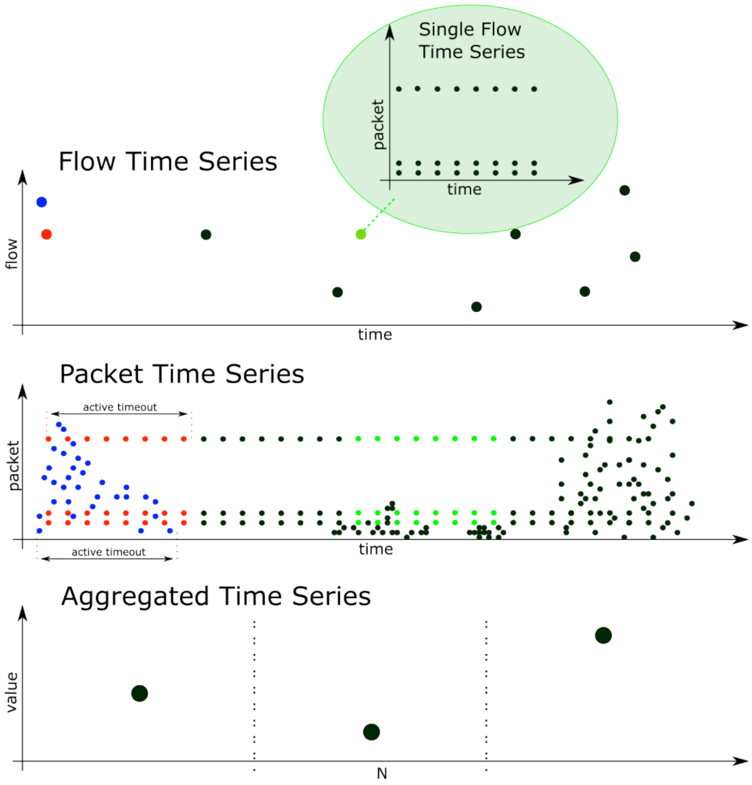

Figure 3.2: Relationship between Packet Time Series, Flow Time Series, Single Flow Time Series, and Evenly Sampled Time Series.

3.2 Network Traffic Classification

Network traffic monitoring provides information about activities in a computer network—an essential insight for maintaining the service and its security. As the technology evolves, a classical approach using *Deep Packet Inspection* (*DPI*) is no longer feasible due to the increased privacy protection using encryption. Additional security features, such as the RFC draft *Encrypted Server Name Indication (ESNI)* [81], which encrypts even domain names, forces the development of new ways of monitoring and analysis to detect network threats and malicious activities.

3.2.1 IP flow Classification

Network traffic monitoring using IP flows is used to handle the current challenge of analyzing encrypted network communication. Nevertheless, the packet aggregation into flow records naturally causes information loss. Furthermore, the SPLT sequence is limited by exporting volumetric values for the first n packets. Therefore, the thesis is focused on using time series analysis to obtain novel features for network traffic classification.

In thesis approach, we create time series from packets within a flow—the series payload sizes in bytes with the corresponding transmission timestamp to create a time series, i.e. Single Flow Time Series as was described in the previous section.

The time series analysis is used to obtain 69 features, which are exported as novel extended IP flow for the purpose of classification. The features can be organized into five categories: 1) statistical, 2) time-based, 3) frequency-based, 4) distribution-based, and 5) behavioral. Some of proposed features for network classification were already used for classification in other fields of science, such as music classification [96, 87, 56]. The detailed description with mathematical equations of the whole feature set is published on the Zenodo platform [A.10].

Statistical-based features The first set of features is based on statistical evaluation of the sequence of observation $\{X_n\}$ of the Single Flow Time Series. The idea is a statistical description of data point deviation, i.e., statistical deviation of the packets' payload lengths. Examples of statistical-based features are Mean, Median, Skewness, Kurtosis, and Entropy.

Time-based features The time-based features describe the time axis of the unevenly-spaced time series $\{x_n\}$. A sequence of relative times $\{rt_n\} = t_i - t_0, i \in \{1, \ldots, n\}$, i.e., time from the beginning of a flow is used for computation time-based features. Additionally, the sequence of time differences $\{dt_{n-1} = t_{i+1} - t_i, I \in \{1, \ldots, n-1\}\}$, i.e., time spaces between packets, is also used. Examples of statistical-based features are Duration, Mean of relative times, and Mean of time differences.

Distribution-based features The set of distribution-based features that are exported in the extended flow describes the distribution of data points in the Single Flow Time

Series $\{x_n\}$. Examples of statistical-based features are Hurst exponent, Benford's law and Normal distribution.

Frequency-based features The idea of frequency-based features is to transform time series into the frequency domain and analyze it. Based on recent research [94, 105, 34], the frequency domain has several advantages over the time-domain. Frequency domain analysis is particularly useful for analyzing periodic behaviors because it allows analysis of the individual frequency components, and can be used to compare the frequency content of different time series.

Frequency domain analysis 1) allows for a more compact representation of a time series, 2) is particularly useful for analyzing periodic behaviors because it allows analysis of the individual frequency components, 3) can be used to compare the frequency content of different time series, that can be useful for identifying similarities or differences between time series and common features or patterns in a set of time series, 4) can filter out unwanted frequency components from a time series that is useful for eliminating noise or other unwanted artifacts from a time series, and lastly 5) can help to identify the underlying sources of variation in a time series.

Since the Single Flow Time Series are unevenly spaced, the Lomb-Scargle (LS) periodogram [100] must be used to transform the time series into a frequency domain. LS was originally developed for unevenly spaced time series in astrophysics.

Examples of statistical-based features are Spectral bandwidth, Spectral centroids, Spectral spread, and Spectral kurtosis.

Behavior-based features The behavior-based features are focused on describing the specific set of behaviors of the Single Flow Time Series. Examples of statistical-based features are Significant spaces, Switching ratio, Transients, and Periodicity.

The thesis explores multiple publicly available datasets previously used or published in the network traffic classification domain. Nevertheless, a lot of datasets consist of already precomputed features and do not contain raw packet-based data, which is necessary for feature extraction based on time-series analysis. Thus, the thesis considered mainly the datasets where raw packet captures (PCAP files) were available. Together is selected 15 well-known network datasets that are written with best-performing related work and selected classification task in Table 3.1 and processed with feature extraction.

Case 1: Classification based on Time Series Analysis of Single Flow Time Series The features were evaluated by creating a novel classifier for each concerned network classification task. On most of the binary classification problems, the novel feature set achieved similar or better performance than the best-performing previous work. Moreover, approach outperformed eight related works significantly (by more than 1%). However, on the TOR detection problem, a worse F1-score than the best classifier is obtained.

Table 3.1: List of binary and multiclass tasks and datasets for binary and multiclass classification.

Binary classification		**Multiclass classification**	
Task	**Dataset**	**Task**	**Dataset**
Botnet det.	CTU-13 [32]	Botnet class.	CTU-13 [32]
Brute-force det.	HTTPS Brute-force [60]	IoT mal. class.	Edge-IIoTset [27]
Mining det.	CESNET-MINER22 [74]		TON_IoT [68]
DNS malware det.	CIC-Bell-DNS [64]	IDS class.	CIC-IDS-2017 [89]
DoH det.	CIC-DoH-Brw [67]		UNSW-NB15 [69]
	DoH-Real-World [47]	TOR class.	ISCX-Tor-2016 [55]
DoS attack det	Bot-IoT [52]	VPN class	ISCX-VPN-2016 [24]
IoT malware det.	IoT-23 [31]		VNAT [49]
	Edge-IIoTset [27] TON_IoT [68]		
Intrusion det.	CIC-IDS-2017 [89] UNSW-NB15 [69]		
TOR det.	ISCX-Tor-2016 [55]		
VPN det.	ISCX-VPN-2016 [24] VNAT [49]		

Moreover, the approach also outperformed most of the best-performing classifiers. Specifically, in five out of eight cases, the approach achieved more than a 1% classification performance increase. However, in two cases, the approach observed a slight decrease—TON_IoT and IDS-UNSW cases.

The proposed method and feature set were evaluated on 23 network classification tasks using 15 publicly available and well-known network traffic datasets which are often used in recent research. All the collected datasets were processed to compute the proposed time series features that were published for any further research by the scientific community.

Case 2: Classification based on the NetTiSA flow The novel extended IP flow with these 69 features inside it cannot be deployed on large or ISP infrastructures. Therefore, the thesis propose a follow-up approach called NetTiSA flow which is designed for classification on high-speed ISP networks.

The thesis proposed a novel extended IP flow called NetTiSA flow built on Time Series Analysis of Single Flow Time Series. The NetTiSA flow contains 13 features based on statistics of payload lengths, statistics of transmission times. These features are computed on the fly, thus, it is possible to process IP flow with an unlimited number of packets. Moreover, an additional seven features can be computed from NetTiSA and traditional flow features, resulting in 20 features called Enhanced NetTiSA. The purpose of additional features is to improve the classification performance.

The list of features that are computed from packets inside the IP flow exported using computation on the fly, is: *Mean*, *Min*, *Max*, *Standard deviation*, *Root mean square*, *Average dispersion*, *Kurtosis*, *Mean of relative times*, *Mean of time differences*, *Min of time differences*, *Max of time differences*, and *Time distribution*. Moreover, the list of features that are computed from the NetTiSA flow on the collector is: *Max minus min*, *Percent deviation*, *Variance*, *Burtiness*, *Coefficient of variation*, and *Directions*.

The usability of the Enhanced NetTiSA feature set was thoroughly evaluated from three main perspectives: 1. Discriminative performance, 2. Flow telemetry size and bandwidth requirements, and 3. Feature computational overhead.

The discriminative performance of the Enhanced NetTiSA feature set was evaluated on 25 network classification tasks using 15 publicly available and well-known network traffic datasets. These datasets were used to train and evaluate ML models and compare their performance to the best-performing classifiers from related works. In almost all cases, a model using the Enhanced NetTiSA feature set gives similar or better results than the best-performing previous proposal for the corresponding task. This confirms the excellent discriminative performance and universality of the proposed features.

At the same time, the size of the NetTiSA flow extension remains small. Even on a network with 1 million flows per second, flows can be exported via a single 1 Gbps line; as experiments showed, computation of NetTiSA features has only negligible impact on the performance of the flow monitoring probe. However, it can still process 100 Gbps of network traffic. Together, these properties make NetTiSA easy to deploy even in large ISP networks. Indeed, the NetTiSA flow extension is already deployed within the production monitoring infrastructure of the CESNET2 network.

The implementation of NetTiSA computation is open-source as part of the *ipfixprobe* flow exporter. Moreover, the implementation of the feature extraction is developed in the form of a library, which allows fast integration into other flow-monitoring software.

3.2.2 Multi-Flow Classification

Network traffic is usually analyzed using time series for traffic forecasting and anomaly detection. On the contrary, there are many regular connections in the background that are promising for automatic detection; thus, the thesis focuses on the detection of such important connections in network traffic and performs classification based on their characteristics.

The thesis focuses on connections distinguished with periodic behavior, which are significant connections. Therefore, first, network traffic is divided into *Network dependencies*, which was presented earlier in this chapter. Therefore, only the IP flows relevant to a potential periodic behavior after splitting the traffic are obtained. Therefore, Flow time series are created. The thesis finds two types of periodic behaviors in the Flow time series. They are shown in the Figures 3.3 and 3.4.

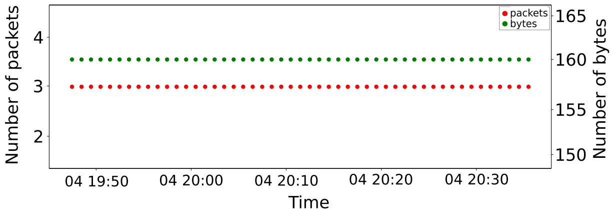

Figure 3.3: Periodic behavior of a time series of the malware Mirai

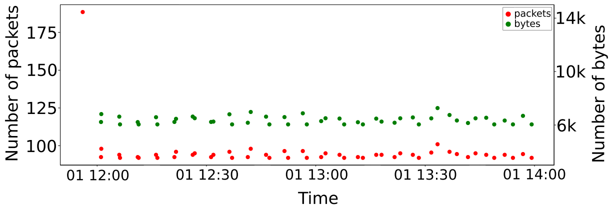

Figure 3.4: Periodic behavior of a time series of the social network Facebook

To detect both types of periodic behaviors in network traffic, mathematical tools are used for the detection of periodicity and periodic behavior in the time series. However, the analysis of unevenly sampled time series must compute not only the values of the data points but also their time, and unfortunately, most time series analysis methods specialize in the evenly sampled time series that appear in most areas. A promising method is a *Lomb-Scargle periodogram* that was defined by Lomb in [58] and Scargle in [85]. It can insert different sine signals into an unevenly sampled time series of periodic behavior and derive frequency and intensity for each, thus creating a periodogram.

The Flow time series is periodic with period p, if there is a statistically significant peak at frequency $f = \frac{1}{p}$. It is therefore necessary to use a statistical significance test on the periodogram. The *Scargle's Cumulative Distribution Function (SCDF)* [30] is suitable for the Lomb-Scargle periodogram, which can be used to determine whether there is any statistically significant peak in the periodogram using a simple formula and also to find out whether a particular peak is statistically significant.

In order to confirm the periodicity of candidate p, an autocorrelation function is used. However, an autocorrelation function does not work with unevenly sampled time series, so the exact timing is neglected.

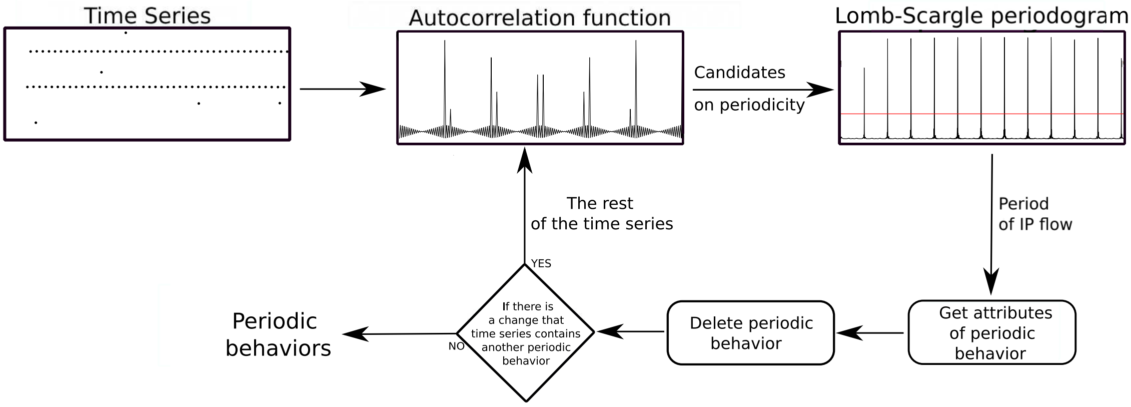

Figure 3.5: Diagram of periodicity detection method

Using the autocorrelation functions computed for each variable of the time series (i.e., number of packets, number of bytes, DiffTimes — a time difference between intermediate data points of time series), histograms of distances between the peaks are created.

The result is a set of candidates for periodicity that significantly appear in all histograms, which can be verified subsequently using the Lomb-Scargle periodogram and the SCDF test. In this particular example, period 2 is chosen as the candidate and a statistically significant peak is then found in the vicinity of the frequency 0.5 and so the period is confirmed.

In general, network dependencies may not always split traffic into ideal time series, thus, there may be multiple periodic behaviors in a single time series. Therefore, there is a need to check whether there is a possibility that the time series does not contain other periodic behavior. The diagram of the whole method is shown in the Fig. 3.5.

Feature engineering is applied to each periodic flow time series. The features are, for example, a *number of flow records* that periodically repeat, a *time period*, a *number of packets*, and a *number of bytes in the flow*.

Case 1: Classification of Applications, Services, and Operating systems A dataset containing 26 thousand samples divided into 61 classification classes was created. As a traffic source for the dataset creation were used multiple publicly available datasets: [28], [91], [22] and [29]. Additionally, the traffic of CESNET2[1], traffic of the Network monitoring laboratory at FIT Czech Technical University in Prague, several home networks, and communication of Android mobile devices were used.

The following traffic categories list the examples of the classes for classification (listed in brackets):

- social networks (Facebook, MS teams, Slack, ...)
- remote storage (Google Drive, OneDrive, Github, ...)

[1] the Czech national research and education network

- updates of operating systems
- antivirus programs (Eset, Avast, Kaspersky, ...)
- game clients (Steam, Epic Games, Uplay, ...)
- network services and protocols (Keep-alive, HTTP2 ping, DNS, ...)
- email browser viewers and clients (Gmail, Outlook, ...)
- multimedia streaming (youtube, itunes, spotify)
- web browsing (firefox, opera)

A Machine Learning pipeline was applied to the created dataset. The best-performing classification algorithm was XGBoost with F1-score 90%. Results of experiments showed that network traffic of the second type of periodicity contains, e.g., social networks communication. This type is harder to recognize. Contrary, system/application level traffic such as keepalive, polling etc. fits to the first type of periodicity. In this case, the classifier is able to recognize each class with higher accuracy.

Case 2: Cryptomining detection Many cryptocurrencies are based on Proof-of-Work (PoW) mechanisms, which consume a lot of electricity and processing power. Thus, the detection of cryptominers is highly necessary for subnetworks where there are servers.

The IP flows for the creation of Flow time series were taken from the CESNET-MINER22 dataset [74], which was created by monitoring the CESNET2[2] network infrastructure by ipfixprobe[3] — open-source flow exporter. The computed FTS created from the CESNET-MINER22 dataset and the datasets of periodic behaviors have been published on Zenodo [A.9].

ML model achieves more than 90% F1-score. Moreover, this approach is suitable for cooperation with the DeCrypto system [73]. Moreover, the DeCrypto system enhanced by periodicity detection achieved the 97.25 % Accuracy, 99.99 % Precision, 92.47 % Recall, and 96.08 % F1-score. That is improvement by 2.95 % Accuracy, 0.001 % Precision, 7.74 % Recall, and 4.37 % F1-score.

Furthermore, the *Dynamic Profile Processing Platform (DP^3)*, available at GitHub[4], is used for the deployment of approach into a high-speed ISP network to enhance the DeCrypto system. The *DP^3* is an open-source data processing platform for maintaining dynamically changing profiles of *entities* represented by sets of attributes of different data types, including various types of time series. It allows the application of custom processing functions over the profiles to enrich, correlate, or otherwise analyze the data to derive new information or detect some events.

[2] The Czech Educational and Science Network

[3] https://github.com/CESNET/ipfixprobe

[4] `https://github.com/CESNET/dp3`

3.3 Anomaly Detection in Network Traffic

In network traffic monitoring, threat detection using supervised learning algorithms is limited to known attacks. Moreover, even if the attack is known the detection is not guaranteed because of a lack of published datasets. Therefore, the supervised learning algorithms can detect only some relatively small part of threats. Therefore, unsupervised detection methods must be implemented to detect unknown or zero-day threats.

Anomaly detection methods usually model a baseline that is used to compare with real observed traffic. Thus, anomaly detection is theoretically possible to deploy in any network. Nevertheless, the thesis focuses on anomaly detection in the second part of the dissertation theme.

3.3.1 Preparation of Real-World Dataset

First of all, the thesis analyzes the current methodologies in the domain. However, most of the datasets for anomaly detection in this domain are old or have not been created for the task of creating a long-term baseline, for example, the month of data. Moreover, a lot of researchers perform binary classification using supervised learning algorithms on the IDS datasets, for example, CIC-IDS-2017 [89], and describe it as anomaly detection. Nevertheless, such a statement completely violates the following mathematical definition of an anomaly:

Definition: *Anomalies being described as rare, unusual, or inconsistent data points that significantly deviate from the majority of the dataset, often detected based on unexpected occurrences within specific data groupings [18].*

Therefore, the thesis focuses on creating a dataset that can support researchers in evaluating anomaly detection algorithms. This dataset may be created from a real-world environment that contains devices with different traffic behaviors. This may help design and evaluate the technique which will work on all types of traffic.

Raw data are captured on the CESNET3 network[5] which is an ISP-level network. This network interconnects all academic institutions in the Czech Republic. However, monitored are also some local networks. Thus, traffic of big NATs, end devices, servers, routers, WiFi devices, some IoT devices, and so on is captured. The CESNET3 network is shown in Figure 3.6.

The CESNET3 network is monitored using the IP flow exporter ipfixprobe[6] which is deployed on multiple monitoring probes on the edge of the CESNET3 network. The Ipfixprobe creates IP flow records which are sent over the infrastructure to the IP flow collector IPFIXcol2[7]. The IP flow records are handled by the modules based on the NEMEA system[8]

[5] Czech Education and Scientific Network

[6] https://github.com/CESNET/ipfixprobe

[7] https://github.com/CESNET/ipfixcol2

[8] https://github.com/CESNET/Nemea

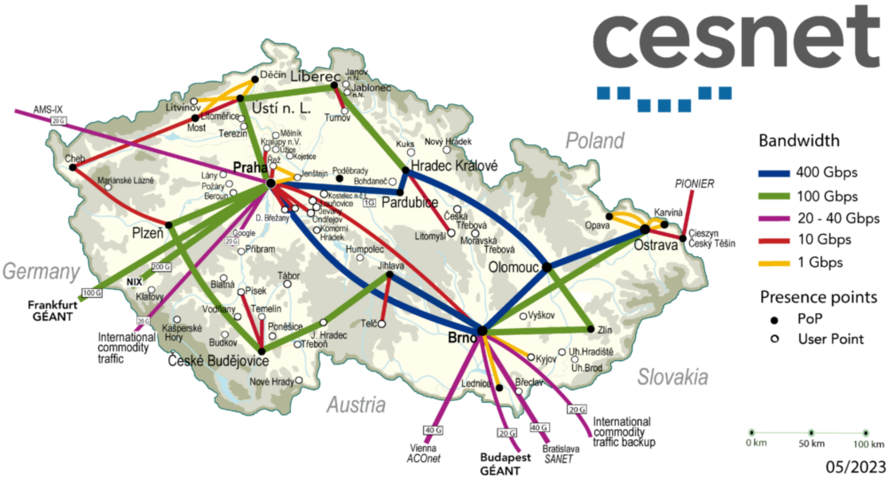

Figure 3.6: Topology of CESNET3 network

[98] which run on the collector.

A NEMEA module called *Aggregator* was created. This module aggregates IP flows into datapoints. These points are sent using the TLS tunnel into TimeSeries Collector which is long-term storage for the time-series-like data. The TimeScaleDB[9] database is used for this purpose. The data is exported from TimeScaleDB after the dataset is collected (after some time period, for example, half a year or one year). Then the annotation of as many IP addresses as possible will be done. Furthermore, IP addresses will be anonymized. The overview of the dataset capturing pipeline is shown in Figure 3.7. This capturing pipeline was deployed into the CESNET3 network on September 25, 2023.

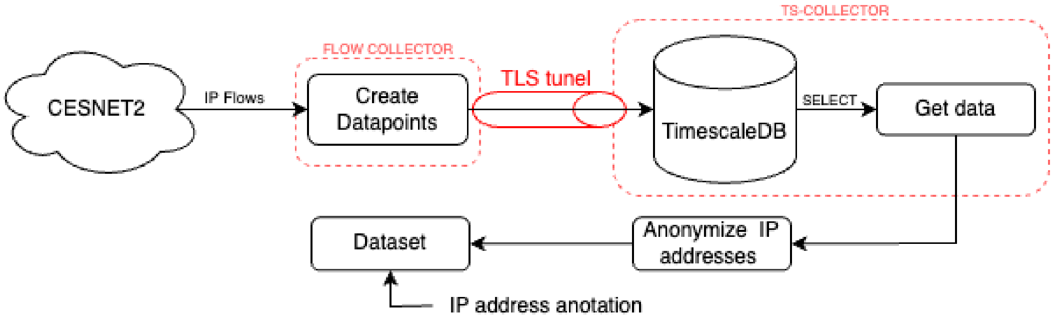

Figure 3.7: Architecture of dataset collection

The *Aggregator* module creates datapoints. The definition of a datapoint is following:

Definition: *Datapoint d_{α,t_i} is a vector of n volumetric metrics computed from the traffic which has specific identifier α and were captured in a time window t_i.*

[9] https://www.timescale.com/

The identifier α defines a time series in which a datapoint belongs. In the case of this dataset, an identifier is an IP address from the list of protected network ranges. However, in the future, experiments with multiple other identifiers, such as a pair of IP addresses, will be conducted.

The n volumetric metrics for each identifier is computed. The volumetric metrics are computed on the fly which means it is not storing raw data in memory. The start and end of metric computation are defined by the size of the aggregation window and the start of data capturing. The aggregation window of size 10 minutes is used (for the bigger window it can be computed from this window using sum or mean). The volumetric metrics are listed below:

- Number of IP flows
- Number of packets
- Number of bytes
- Number of unique destination public IP addresses
- Number of unique destination ASNs
- Number of unique destination countries
- Number of unique destination transport layer ports
- Ratio of TCP/UDP traffic for number of packets
- Ratio of TCP/UDP traffic for number of bytes
- Directions ratio in a matter of number of packets
- Directions ratio in a matter of number of bytes
- Average duration of IP flows
- Average TTL of IP flows

The data collection started on September 25, 2023, which is usually the start of the semester in the Czech Republic. At the moment of writing this manuscript, more than 30 weeks were captured, which can also be described as almost two academic semesters. The dataset now consists of more than 750 thousand time series, i.e., data of 750 thousand IP addresses. The dataset will be published after two semesters of data will be captured fully.

3.3.2 Anomaly Detection System

In this domain, an anomaly detection method is usually evaluated only based on the precision of the model, and the deployability of the model is not considered. However, the deployability of the model is more crucial than precision because it is useless to have the most precise model if the model cannot be deployed in a real-world network. Thus, in this thesis is designed an Anomaly Detection System (ADS) for deploying anomaly detection models in a real-world environment.

This system is designed to be able to process huge amounts of data and a new model can be easily integrated into the ADS system without changing the system code. The ADS consists of several modules that cooperate in performing anomaly detection. Part of the ADS system was introduced in the previous subsection. It is *Aggregator* module and *TimeScaleDB*, i.e., this part of the ADS system is used for capturing the CESNET-AD-2023/2024 dataset.

The second part of the ADS system are *Trainers* and *Detectors* modules. The *Trainers* module dynamically imports enabled models from configuration and runs the training function of each model at the start of the day. The training function of a model usually trains baseline from historical data in a database and predicts future datapoints. The *Detectors* module runs the detection function of each model. The detecting function of a model usually compares observed datapoints with predicted datapoints after the end of the aggregation window, and if the observed datapoint significantly differs from the predicted datapoint the anomaly is detected.

Therefore, adding a new model into the ADS system is straightforward. Just write training and detection functions and add the model to the configuration.

Moreover, the *Aggregation*, *Trainers*, and *Detectors* can be easily run several times to support anomaly detection on multiple aggregation windows, for example, the ADS system can be run with the following three aggregation intervals: 10 minutes, one hour, and one day.

Furthermore, the ADS system allows combining alerts from all volumetric metrics and all aggregation windows for each IP address in the module called *Alerts Combiner*. Therefore, there is a research challenge of evidence combination. The overall architecture of the ADS system is shown in Figure 3.8.

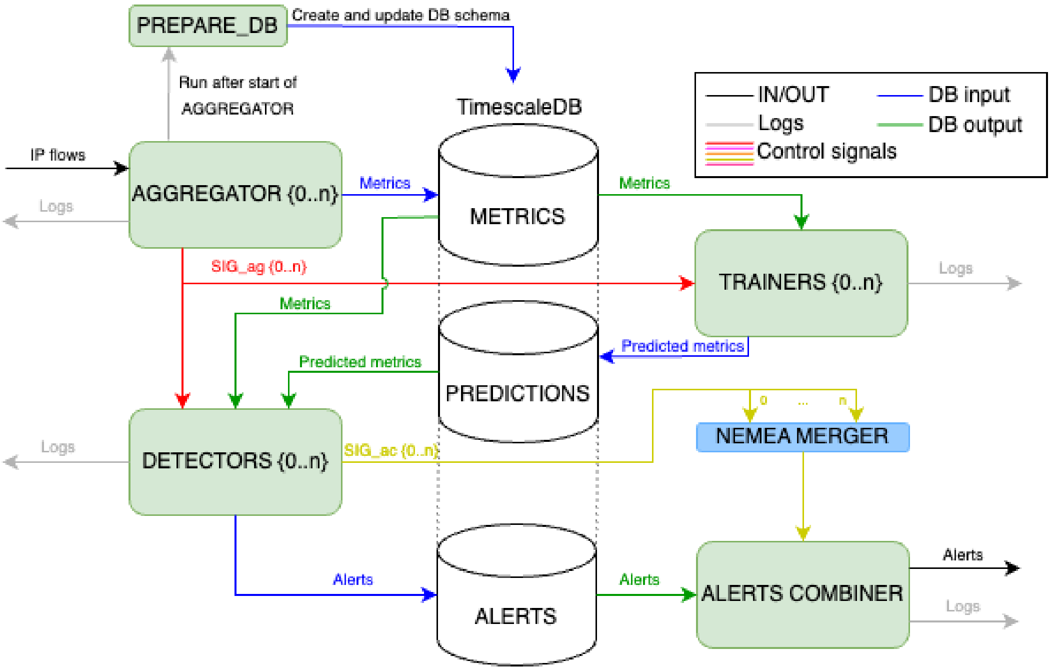

Figure 3.8: Architecture of the ADS system

The most common anomaly detection models and the most promising novel models from related literature will be evaluated using the developed ADS system deployed in the CESNET3 network. Thus, it will bring insight into the applicability of anomaly detection models to real-world networks.

Chapter 4

Preliminary Results

This chapter describes the results of the thesis research which were achieved during the development of the proposed solution.

4.1 Time Series from Network Traffic

To support thesis research statements the definition types of time series that can be created from network traffic as described in Section 3.1. The first focus was on unevenly spaced time series from network traffic. Thus, several types of unevenly spaced time series from network traffic were defined and their properties were evaluated. Paper [A.5] called *"Unevenly spaced time series from network traffic"* proposes an Unevenly Spaced Time Series as a feasible representation of network traffic in time with several benefits for analysis. The paper concerns several types of Unevenly Space Time Series — Packet Time Series, Flow Time Series, and Single Flow Time Series. To evaluate the properties of the concerned types of series, a dataset was created by capturing traffic on a real ISP network with half a million users and over 35 million network traffic time series. According to the results, the relevant series are suitable for modeling network traffic and allow automatic processing without needing trend and seasonal analysis. Thus, the proposed USTS are suitable for network traffic analysis and the detection of security threats and can form a basis for future network security detectors.

4.2 Classification based on Periodic IP flows

As mentioned in Section 3.1, a time series of multiple flows that were likely generated by the same process can be created. they are referred to as Flow Time Series in the research and thesis. Analysis of these time series shows that some of these time series contain strong periodic patterns. The thesis focus research on the detection and description of periodic patterns in two publications [A.4, A.3]. Both of these publications use Flow Time Series and detection of periodic patterns in common. However, their methodology, model, and detection problem differ. Both publications are deeply described below:

4.2.1 Model 1: Applications, Services, and Operating systems

Paper [A.4] called *"Network traffic classification based on periodic behavior detection"* deals with the detection of periodic behavioral patterns of communication by autocorrelation function and Lomb-Scargle periodogram. After the detection, the periodic pattern is described by a set of parameters, for example, time period, number of packets and bytes of periodically repeating flow, and intervals of packets and bytes for periodic behavior. The revealed characteristics of the periodic behavior can be further exploited to recognize particular applications, services, and operating systems. On the created dataset were performed experiments, and a machine learning classifier based on XGBoost performed the best in experiments, reaching 90% F1-score.

4.2.2 Model 2: Cryptomining

Subsequent experiments show that model is too complex and that the computational requirements of the model prevent deployment. Therefore, paper [A.3] called *"Enhancing DeCrypto: Finding Cryptocurrency Miners based on Periodic Behavior"* is focusing on enhancing Model 1 by decreasing computational complexity and defining a larger set of features. Furthermore, a novel model analyzes the long-term periodic behavior of cryptocurrency miners communicating in computer networks. A novel method was proposed for cryptominer detection using specially designed periodicity features which include statistical features and frequency-based features. Altogether with the Machine Learning technique, the resulting system achieves high-precision performance. Furthermore, the approach enhances a flow-based cryptominers detection system, DeCrypto, to further improve its reliability and feasibility for high-speed networks.

4.3 Classification of IP flows

The next research theme was classifying each IP flow, one of the most dominant domains in network traffic monitoring. The approach uses time series analysis of time series called Single Flow Time Series (described in Section 3.1). This time series usually contains specific behavior of the generating process that can be used for detection, i.e., creating new traffic representations. As described in the related works section, the SPLT sequence is a time series of first n packets that has great precision with using Neural Networks. Approach differs by using time series analysis to get properties of behavior. Two papers were published on using this approach to the detection of security threats that follow each other.

4.3.1 Model 1: Time Series Analysis of Single Flow Time Series

Paper [A.2] called *"Network Traffic Classification based on Single Flow Time Series Analysis"* proposes a novel flow extension for traffic features based on the time series analysis of the Single Flow Time series. Paper [A.2] proposes 69 universal features based on the statistical analysis of data points, time domain analysis, packet distribution within the flow timespan, time series behavior, and frequency domain analysis. Paper [A.2] demonstrated the usability and universality of the proposed feature vector for various network traffic classification tasks using 15 well-known publicly available datasets. Evaluation shows that the novel feature vector achieves classification performance similar or better than related works on both binary and multiclass classification tasks. In more than half of the evaluated tasks, the classification performance increased by up to 5 %.

4.3.2 Model 2: NetTiSA flow

Model 1 is universal and archives an increase in performance on most of the classification tasks. However, the computational and memory cost makes Model 1 unable to be deployed in high-speed networks. Therefore, paper [A.1] called *"NetTiSA: Extended IP Flow with Time-series Features for Universal Bandwidth-constrained High-speed Network Traffic Classification"* builds on previous work [A.2] and proposes a novel extended IP flow called NetTiSA (Network Time Series Analysed) flow. In this paper, features that can be computed online in exporter are picked up without saving the Single Flow Time Series into the RAM memory of the network probe. By thoroughly testing 25 different network traffic classification tasks, paper [A.1] shows the broad applicability and high usability of NetTiSA flow. For practical deployment, the paper [A.1] also considers the sizes of flows extended by NetTiSA features and evaluates the performance impacts of their computation in the flow exporter. The novel features proved to be computationally inexpensive and showed excellent discriminatory performance. The trained machine learning classifiers with proposed features mostly outperformed the state-of-the-art methods. NetTiSA finally bridges the gap and brings universal, small-sized, and computationally inexpensive features for traffic classification that can be scaled up to extensive monitoring infrastructures, bringing the machine learning traffic classification even to 100 Gbps backbone lines.

4.4 Datasets and Source codes

To support research statements and results, this research supported the open science movement; thus, all input data are publicly available for the research community, and also source codes of experiments are publicly available for replication of results.

All the published datasets, including citations and statistical[1] overview, is shown in Table 4.2.

The publicly available code include experiment for papers [A.1, A.2, A.3, A.5] and plugins to open-source flow exporter ipfixprobe[2]. These plugins are based on papers [A.1, A.2].

[1] Description of each column is defined by the Zenodo platform in `https://help.zenodo.org/faq/#statistics`

[2] https://github.com/CESNET/ipfixprobe

Table 4.1: Detailed statistics for published datasets as of April 25, 2024.

Dataset name	Publish	Views	Downloads	Size [GB]
CESNET-USTS23: a benchmark dataset of Unevenly spaced time series from network traffic	May 11, 2023, [A.8]	127	151	3.7
CESNET-MINER22-TS: Periodic Behavior Features of Cryptomining Communication	June 13, 2023, [A.9]	193	33	1.6
Network traffic datasets created by Single Flow Time Series Analysis	Jun 14, 2023, [A.10]	1163	1301	111.0
Network traffic datasets with novel extended IP flow called NetTiSA flow	Aug 30, 2023, [A.11]	277	630	28.3

Table 4.2

Description	Link to Github
Experiments for the paper Unevenly Spaced Time Series from Network Traffic	`https://github.com/koumajos/USTS`
Experiments for the paper Enhancing DeCrypto: Finding Cryptocurrency Miners based on Periodic Behavior	`https://github.com/koumajos/EnhancedDeCrypto`
TSA of SFTS as a plugin for Ipfixprobe flow exporter	`https://github.com/koumajos/ipfixprobe_tsa_sfts`
Classification based on TSA of SFTS	`https://github.com/koumajos/ClassificationBasedOnSFTS`
NetTiSA flow as a plugin for Ipfixprobe flow exporter	`https://github.com/CESNET/ipfixprobe`
Classification based on NetTiSA flows	`https://github.com/koumajos/Classification_by_NetTiSA_flow`

Chapter 5

Conclusions

The increasing trend in traffic encryption causes problems for network security traffic monitoring. Threat actors can leverage privacy-preserving technologies to hide their malicious activities. Therefore, there is an urgent need to develop novel methods and technologies capable of finding security threats while preserving the users' privacy.

The goal of this report is to introduce possible approaches based on time series analysis that solve problems in the network traffic monitoring domain caused by increasing traffic encryption. The concept of time series analysis and state-of-the-art of time series analysis in network traffic monitoring was described in Chapter 2. The surveyed studies challenge the privacy aspects of security protocols, demonstrating the possibility that time series analysis can be used to get a view of encrypted traffic.

Even though the number of studies focusing on the classification of encrypted traffic using time series is relatively high, their main shortcoming is that they usually use time series only as input into neural networks and do not focus on more complex time series analysis. Therefore, I concentrate my research on more complex time series analysis in two domains of classification. First is the IP flow classification, where I propose novel feature vectors for classification based on feature engineering, which uses time series analysis. Second is the multi-flow classification, where I propose a novel classification method based on periodic behavior detection.

Moreover, I also do preliminary experiments and implementation to continue my research in the domain of anomaly detection. I believe that dataset CESNET-AD-2023/2024 will bring insight into the actual precision and deployability of current anomaly detection methods and also bring new challenges caused by complex real-world environments that novel methods may overcome. Moreover, Anomaly Detection System deployed in the CESNET3 network will serve as an ideal environment for testing the deployability of anomaly detection methods.

Based on the analysis of state-of-the-art and preliminary results, the doctoral thesis will be focused on three main topics described in the following section.

5.1 Proposed Doctoral Thesis

Title of the future dissertation thesis:

Threat Detection in Network Traffic using Time Series Analysis

This report suggests the following primary topics addressed in the future dissertation thesis:

5.1.1 Supervised Classification using Time Series Analysis

Network monitoring plays a crucial role in the overall computer security management. Compared to protections (such as AntiVirus software) deployed on the end devices, the network-based intrusion detection and prevention systems can protect infrastructure against users' sloppiness, policy violations, or (at worst) intentional attacks from the inside. **However, maintaining network security has become increasingly challenging in recent years due to mass traffic encryption and consequent reduced visibility.** The encryption of TLS certificates by TLS1.3 [80], deployment of encrypted DNS [39], or Encrypted Client Hello proposal [81] removed the few-remaining information essential for effective threat detection. Therefore, classification of network traffic based on remaining information such as packet length, packet time, and others is required. **However, building an accurate model that will universally work based on it is challenging.**

Objectives of the future thesis: Therefore, I focused on the classification of network traffic and brought novel methods of detection of security threats. Thus, the first half of the future doctoral thesis will stand on network traffic classification using time series analysis. It is possible to extend my work with testing approaches on the next tasks, such as classification TLS or QUIC services using the NetTiSA flow or botnet detection using the flow time series and periodic behavior detection.

5.1.2 Unsupervised Classification using Time Series Analysis

As cyber threats evolve in complexity, effective anomaly detection systems in network traffic become essential for early detection of potential threats such as malware infections, data breaches, or denial of service attacks, and for maintaining compliance with strict data security regulations. Anomaly detection also aids in identifying unexpected network behaviors that can impact performance and reliability, and is crucial in protecting data privacy by identifying unauthorized data access or abnormal data usage patterns.

The primary challenges in anomaly detection research include managing high false positive rates where normal activities are often misclassified as anomalies, adapting to dynamic and evolving threats that conventional models may not quickly recognize, and deploying these systems into high-speed network environments where they **must operate efficiently without slowing down network traffic**. Additionally, handling the volume, velocity, and variety of network data poses **significant scalability and real-time processing challenges**. There is also a **scarcity of labeled data for training sophisticated models**, and the complexity of modern network environments complicates the development of universally applicable anomaly detection systems.

Objectives of the future thesis: Therefore, I will also focus on unsupervised classification of network traffic which is mainly used for anomaly detection and network traffic prediction. Anomaly detection methods are in high demand in this domain. I will evaluate the most common anomaly detection models and the most promising novel models from related literature using the developed ADS system deployed in the CESNET3 network. Thus, I will bring insight into the applicability of anomaly detection models to real-world networks.

5.1.3 Time Series based Datasets of Network Traffic

The rise of using machine learning in the monitoring of network traffic creates a high demand for datasets. Most of the published datasets by now are created in a laboratory. However, for example, for anomaly detection that can be deployed into real networks, the research community must have the datasets of real network traffic. In network traffic monitoring exists a challenge to building methods for classification based on longer windows than a set of packets or one IP flow. These methods can be used to cooperate with recent methods to increase the precision of the overall system in their detection. However, **the time series based datasets are missing**.

Moreover, **most of the datasets for anomaly detection in this domain are old or short** — they have not been created for the task of creating a long-term baseline, for example, a month of data.

Objectives of the future thesis: Therefore, I will focus on creating datasets that can support researchers in evaluating anomaly detection algorithms. These datasets will be created from a complex real-world environment. Datasets with these properties are highly demanded by now but missing.

Bibliography

[1] Amaia Abanda, Usue Mori, and Jose A. Lozano. A review on distance based time series classification. *Data Mining and Knowledge Discovery*, 33(2):378–412, 2019.

[2] Shubair A Abdullah and Amaal S Al-Hashmi. Tisefe: Time series evolving fuzzy engine for network traffic classification. *International Journal of Communication Networks and Information Security*, 10(1):116–124, 2018.

[3] Giuseppe Aceto, Domenico Ciuonzo, Antonio Montieri, and Antonio Pescapé. Distiller: Encrypted traffic classification via multimodal multitask deep learning. *Journal of Network and Computer Applications*, 183:102985, 2021.

[4] Rakesh Agrawal, Ramakrishnan Srikant, et al. Fast algorithms for mining association rules. In *Proc. 20th int. conf. very large data bases, VLDB*, volume 1215, pages 487–499. Santiago, Chile, 1994.

[5] Theyazn HH Aldhyani et al. Intelligent hybrid model to enhance time series models for predicting network traffic. *IEEE Access*, 8, 2020.

[6] Blake Anderson and David McGrew. Identifying encrypted malware traffic with contextual flow data. In *Proceedings of the 2016 ACM workshop on artificial intelligence and security*, pages 35–46, 2016.

[7] Azeem Aqil, Karim Khalil, Ahmed O.F. Atya, Evangelos E. Papalexakis, Srikanth V. Krishnamurthy, Trent Jaeger, K. K. Ramakrishnan, Paul Yu, and Ananthram Swami. Jaal: Towards network intrusion detection at isp scale. In *Proceedings of the 13th International Conference on Emerging Networking EXperiments and Technologies*, CoNEXT '17, page 134–146, New York, NY, USA, 2017. Association for Computing Machinery.

[8] Goethem Arthur Van, Frank Staals, Maarten Löffler, Jason Dykes, and Bettina Speckmann. Multi-granular trend detection for time-series analysis. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):661–670, 2017.

[9] Lei Bai, Lina Yao, Salil S. Kanhere, Xianzhi Wang, and Zheng Yang. Automatic device classification from network traffic streams of internet of things. In *2018 IEEE 43rd Conference on Local Computer Networks (LCN)*, pages 1–9, 2018.

[10] Kurt Barbe, Rik Pintelon, and Johan Schoukens. Welch method revisited: Nonparametric power spectrum estimation via circular overlap. *IEEE Transactions on Signal Processing*, 58(2):553–565, 2010.

[11] M. S. Bartlett. Periodogram analysis and continuous spectra, Jun 1950.

[12] Jarosław Bernacki and Grzegorz Kołaczek. Anomaly detection in network traffic using selected methods of time series analysis. *International Journal of Computer Network and Information Security*, 2015.

[13] Alina Beygelzimer, Emre Erdogan, Sheng Ma, and Irina Rish. Statistical models for unequally spaced time series. 04 2005.

[14] Thitima Booranawong and Apidet Booranawong. Double exponential smoothing and holt-winters methods with optimal initial values and weighting factors for forecasting lime, thai chili and lemongrass prices in thailand. *Engineering and Applied Science Research*, 45:32–38, 02 2018.

[15] George EP Box, Gwilym M Jenkins, Gregory C Reinsel, and Greta M Ljung. *Time series analysis: forecasting and control.* John Wiley & Sons, 2015.

[16] Mohammad Braei and Sebastian Wagner. Anomaly detection in univariate time-series: A survey on the state-of-the-art, 2020.

[17] Sherree Buchenroth and Robert Jennings. A descriptive analysis of the time series behavior of financial analysts' earnings forecasts. *Quarterly Journal of Business and Economics*, 26(3):22–41, 1987.

[18] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM computing surveys (CSUR)*, 41(3):1–58, 2009.

[19] Chris Chatfield and Haipeng Xing. *The analysis of time series: an introduction with R.* Chapman and hall/CRC, 2019.

[20] Benoit Claise. Cisco Systems NetFlow Services Export Version 9. *RFC*, 3954:1–33, 2004.

[21] Benoit Claise, Brian Trammell, and Paul Aitken. Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information. *RFC*, 7011:1–76, 2013.

[22] G. Creech and J. Hu. *ADFA IDS Dataset, University of Arizona Artificial Intelligence Lab, AZSecure-data.* Director Hsinchun Chen, November 2016. `http://www.azsecure-data.org/`.

[23] Henrique Dornel, EDS Christo, Kelly Alonso Costa, and DPM Souza. Traffic forecasting for monitoring in computer networks using time series. *Int. J. Adv. Eng. Res. Sci*, 6(7), 2019.

[24] Gerard Draper-Gil et al. Characterization of Encrypted and VPN Traffic Using Time-related. In *Proceedings of the 2nd international conference on information systems security and privacy (ICISSP)*, pages 407–414, 2016.

[25] Mohamed G Elfeky, Walid G Aref, and Ahmed K Elmagarmid. Periodicity detection in time series databases. *IEEE Transactions on Knowledge and Data Engineering*, 17(7):875–887, 2005.

[26] Meisam Eslahi, M. S. Rohmad, Hamid Nilsaz, Maryam Var Naseri, N.M. Tahir, and H. Hashim. Periodicity classification of http traffic to detect http botnets. In *2015 IEEE Symposium on Computer Applications and Industrial Electronics (ISCAIE)*, pages 119–123, 2015.

[27] Mohamed Amine Ferrag et al. Edge-iiotset: A new comprehensive realistic cyber security dataset of iot and iiot applications: Centralized and federated learning, 2022.

[28] Romain Fontugne et al. Mawilab: Combining diverse anomaly detectors for automated anomaly labeling and performance benchmarking. In *ACM CoNEXT '10*, December 2010.

[29] Student Union for Electrical Engineering (Fachbereichsvertretung Elektrotechnik) at Ulm University and Philipp Hinz. *2017 SUEE data set*. `https://github.com/vs-uulm/2017-SUEE-data-set`, Accessed: 2022-06-24.

[30] Fabio Frescura, Chris Engelbrecht, and B. Frank. Significance tests for periodogram peaks. 2007.

[31] Sebastian Garcia et al. IoT-23: A labeled dataset with malicious and benign IoT network traffic, January 2020. More details here https://www.stratosphereips.org /datasets-iot23.

[32] S. García et al. An Empirical Comparison of Botnet Detection Methods. *Computers & Security*, 45:100–123, 2014.

[33] F Godtliebsen, LR Olsen, and J-G Winther. Recent developments in statistical time series analysis: Examples of use in climate research. *Geophysical Research Letters*, 30(12), 2003.

[34] Ralph Haberl et al. Comparison of Frequency and Time Domain Analysis of the Signal-averaged Electrocardiogram in Patients With Ventricular Tachycardia and Coronary Artery Disease: Methodologic Validation and Clinical Relevance. *Journal of the American College of Cardiology*, 12(1):150–158, 1988.

[35] Mackenzie Haffey, Martin Arlitt, and Carey Williamson. Modeling, analysis, and characterization of periodic traffic on a campus edge network. In *2018 IEEE 26th*

International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS), pages 170–182, 2018.

[36] James D Hamilton. *Time series analysis.* Princeton university press, 2020.

[37] Jiawei Han, Guozhu Dong, and Yiwen Yin. Efficient mining of partial periodic patterns in time series database. In *Proceedings 15th International Conference on Data Engineering.* IEEE, 1999.

[38] Shital P Hatkar, SH Kadam, and AH Syed. Analysis of various periodicity detection algorithms in time series data with design of new algorithm. *International Journal of Computer Applications Technology and Research*, 3:229–239, 2014.

[39] Paul E. Hoffman and Patrick McManus. DNS Queries over HTTPS (DoH). RFC 8484, October 2018.

[40] Rick Hofstede, Pavel Čeleda, Brian Trammell, Idilio Drago, Ramin Sadre, Anna Sperotto, and Aiko Pras. Flow monitoring explained: From packet capture to data analysis with netflow and ipfix. *IEEE Communications Surveys & Tutorials*, 16(4):2037–2064, 2014.

[41] Jawad Mahmud Hoque, Gregory D Erhardt, David Schmitt, Mei Chen, Ankita Chaudhary, Martin Wachs, and Reginald R Souleyrette. The changing accuracy of traffic forecasts. *Transportation*, 49(2):445–466, 2022.

[42] Jawad Mahmud Hoque, Gregory D Erhardt, David Schmitt, Mei Chen, and Martin Wachs. Estimating the uncertainty of traffic forecasts from their historical accuracy. *Transportation research part A: policy and practice*, 147:339–349, 2021.

[43] Neminath Hubballi and Deepanshu Goyal. Flowsummary: Summarizing network flows for communication periodicity detection. In *Pattern Recognition and Machine Intelligence*, 2013.

[44] Ren-Hung Hwang, Min-Chun Peng, Chien-Wei Huang, Po-Ching Lin, and Van-Linh Nguyen. An unsupervised deep learning model for early network traffic anomaly detection. *IEEE Access*, 8:30387–30399, 2020.

[45] R.J. Hyndman and G. Athanasopoulos. *Forecasting: Principles and Practice.* 2014.

[46] Hassan Ismail Fawaz, Germain Forestier, Jonathan Weber, Lhassane Idoumghar, and Pierre-Alain Muller. Deep learning for time series classification: a review. *Data Mining and Knowledge Discovery*, 33(4):917–963, 2019.

[47] Kamil Jeřábek et al. Collection of datasets with DNS over HTTPS traffic. *Data in Brief*, 42:108310, 2022.

[48] Ming JIANG, Chun-ming WU, Min Zhang, and Da-min HU. Research on the comparison of time series models for network traffic prediction. *ACTA ELECTONICA SINICA*, 37(11):2353, 2009.

[49] Steven Jorgensen et al. Extensible Machine Learning for Encrypted Network Traffic Application Labeling via Uncertainty Quantification. *CoRR*, abs/2205.05628, 2022.

[50] Sangjoon Jung, Chonggun Kim, and Younky Chung. A prediction method of network traffic using time series models. In *International Conference on Computational Science and Its Applications*, 2006.

[51] Huang Kai, Qi Zhengwei, and Liu Bo. Network anomaly detection based on statistical approach and time series analysis. In *2009 International Conference on Advanced Information Networking and Applications Workshops*, pages 205–211. IEEE, 2009.

[52] Nickolaos Koroniotis et al. Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset. *Future Gener. Comput. Syst.*, 100:779–796, 2019.

[53] Peter Kromkowski et al. Evaluating statistical models for network traffic anomaly detection. In *2019 systems and information engineering design symposium (SIEDS)*, pages 1–6. IEEE, 2019.

[54] R. Kumari, Sheetanshu, M. K. Singh, R. Jha, and N.K. Singh. Anomaly detection in network traffic using k-mean clustering. In *2016 3rd International Conference on Recent Advances in Information Technology (RAIT)*, pages 387–393, 2016.

[55] Arash Habibi Lashkari et al. Characterization of Tor Traffic using Time based Features. In *ICISSP 2017*, pages 253–262. SciTePress, 2017.

[56] Alexander Lerch. *An Introduction to Audio Content Analysis: Applications in Signal Processing and Music Informatics*. Wiley-IEEE Press, 2012.

[57] Ta-Hsin Li. Laplace periodogram for time series analysis. *Journal of the American Statistical Association*, 103(482):757–768, 2008.

[58] N.R. Lomb. Least-squares frequency analysis of unequally spaced data. 1976.

[59] Jan Luxemburk and Tomáš Čejka. Fine-grained tls services classification with reject option. *Computer Networks*, 220:109467, 2023.

[60] Jan Luxemburk, Karel Hynek, and Tomas Cejka. HTTPS Brute-force dataset with extended network flows, November 2020.

[61] Jan Luxemburk, Karel Hynek, and Tomáš Čejka. Encrypted traffic classification: the quic case. In *2023 7th Network Traffic Measurement and Analysis Conference (TMA)*, pages 1–10. IEEE, 2023.

[62] Qian Ma, Cong Sun, Baojiang Cui, and Xiaohui Jin. A novel model for anomaly detection in network traffic based on kernel support vector machine. *Computers & Security*, 104:102215, 2021.

[63] Rishabh Madan et al. Predicting computer network traffic: a time series forecasting approach using DWT, ARIMA and RNN. In *2018 11th International Conference on Contemporary Computing (IC3)*, 2018.

[64] Samaneh Mahdavifar et al. Classifying Malicious Domains using DNS Traffic Analysis. In *DASC/PiCom/CBDCom/CyberSciTech 2021*, pages 60–67. IEEE, 2021.

[65] W.W. Melek, Ziren Lu, Alex Kapps, and William Fraser. Comparison of trend detection algorithms in the analysis of physiological time-series data. *IEEE transactions on bio-medical engineering*, 52:639–51, 05 2005.

[66] Manuel Méndez, Mercedes G Merayo, and Manuel Núñez. Long-term traffic flow forecasting using a hybrid cnn-bilstm model. *Engineering Applications of Artificial Intelligence*, 121:106041, 2023.

[67] Mohammadreza MontazeriShatoori et al. Detection of DoH Tunnels using Time-series Classification of Encrypted Traffic. In *2020 IEEE Intl Conf: DASC/PiCom/CBDCom/CyberSciTech*, pages 63–70, 2020.

[68] Nour Moustafa. A new distributed architecture for evaluating ai-based security systems at the edge: Network ton_iot datasets. *Sustainable Cities and Society*, 72:102994, 2021.

[69] Nour Moustafa and Jill Slay. Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set). In *2015 military communications and information systems conference (MilCIS)*, pages 1–6. IEEE, 2015.

[70] Eva Ostertagova and Oskar Ostertag. Forecasting using simple exponential smoothing method. *Acta Electrotechnica et Informatica*, 12:62–66, 12 2012.

[71] Dimou Paraskevi, Jan Fajfer, Nicolas Müller, Eva Papadogiannaki, Evangelos Rekleitis, and František Střasák. *Encrypted Traffic Analysis, Use Cases & Security Challenges.* 2020.

[72] Karl Pearson. Notes on the history of correlation. *Biometrika*, 13(1):25–45, 1920.

[73] Richard Plný et al. DeCrypto: Finding Cryptocurrency Miners on ISP Networks. In *NordSec 2022*, volume 13700, pages 139–158. Springer, 2022.

[74] Richard Plný et al. *Datasets of Cryptomining Communication.* Zenodo, October 2022.

[75] Tom Puech, Matthieu Boussard, Anthony D'Amato, and Gaëtan Millerand. A fully automated periodicity detection in time series. In *International Workshop on Advanced Analysis and Learning on Temporal Data*, pages 43–54. Springer, 2020.

[76] Yong Qiao et al. Detecting P2P bots by mining the regional periodicity. *Journal of Zhejiang University SCIENCE C*, 14(9):682–700, Sep 2013.

[77] Yanjun Qin, Haiyong Luo, Fang Zhao, Yuchen Fang, Xiaoming Tao, and Chenxing Wang. Spatio-temporal hierarchical mlp network for traffic forecasting. *Information Sciences*, 632:543–554, 2023.

[78] Suhasini Subba Rao. A course in time series analysis. *Technical Report, Texas A&M University*, 2022.

[79] Faras Rasheed, Mohammed Alshalalfa, and Reda Alhajj. Efficient periodicity mining in time series databases using suffix trees. *IEEE Transactions on Knowledge and Data Engineering*, 23(1):79–94, 2010.

[80] Eric Rescorla. The Transport Layer Security (TLS) Protocol Version 1.3. RFC 8446, August 2018.

[81] Eric Rescorla et al. TLS Encrypted Client Hello. Internet-Draft draft-ietf-tls-esni-16, Internet Engineering Task Force, 2023.

[82] Ganesh Sadasivan et al. Architecture for IP Flow Information Export. *RFC*, 5470:1–31, 2009.

[83] Amir M. Sadeghzadeh et al. Adversarial network traffic: Towards evaluating the robustness of deep-learning-based network traffic classification. *IEEE Transactions on Network and Service Management*, 18(2), 2021.

[84] Łukasz Saganowski and Tomasz Andrysiak. Time series forecasting with model selection applied to anomaly detection in network traffic. *Logic Journal of the IGPL*, 28(4):531–545, 2020.

[85] Jeffrey Scargle. Studies in astronomical time series analysis. ii - statistical aspects of spectral analysis of unevenly spaced data. *The Astrophysical Journal*, 263, December 1982.

[86] Dominik Schatzmann, Wolfgang Mühlbauer, Thrasyvoulos Spyropoulos, and Xenofontas A. Dimitropoulos. Digging Into HTTPS: Flow-based Classification of Webmail Traffic. In Mark Allman, editor, *Proceedings of the 10th ACM SIGCOMM Internet Measurement Conference, IMC 2010, Melbourne, Australia - November 1-3, 2010*, pages 322–327. ACM, 2010.

[87] Eric D. Scheirer and Malcolm Slaney. Construction and Evaluation of a Robust Multifeature Speech/Music Discriminator. In *ICASSP 1997*, pages 1331–1334. IEEE Computer Society, 1997.

[88] Arthur Schuster, Henry Ludwell Moore, and A. E. Douglass. *Periodogram analysis*. 1898.

[89] Iman Sharafaldin et al. Toward generating a new intrusion detection dataset and intrusion traffic characterization. *ICISSp*, 1:108–116, 2018.

[90] Federico Simmross-Wattenberg, Juan Ignacio Asensio-Perez, Pablo Casaseca-de-la Higuera, Marcos Martin-Fernandez, Ioannis A. Dimitriadis, and Carlos Alberola-Lopez. Anomaly detection in network traffic based on statistical inference and $\alpha - stablemodeling.IEEETransactionsonDependableandSecureComputing, 8(4): 494 - -509, 2011.$

[91] Manmeet; Singh Singh, Maninder; Kaur, and Sanmeet. *"10 Days DNS Network Traffic from April-May, 2016", Mendeley Data, V2.* doi: 10.17632/zh3wnddzxy.2.

[92] P. Sonali and D. Nagesh Kumar. Review of trend detection methods and their application to detect temperature changes in india. *Journal of Hydrology*, 476:212–227, 2013.

[93] Cédric St-Onge, Nadjia Kara, Omar Abdel Wahab, Claes Edstrom, and Yves Lemieux. Detection of time series patterns and periodicity of cloud computing workloads. *Future Generation Computer Systems*, 109:249–261, 2020.

[94] A. Suarez et al. Analytical Comparison Between Time- And Frequency-domain Techniques for Phase-noise Analysis. *IEEE Transactions on Microwave Theory and Techniques*, 50(10):2353–2361, 2002.

[95] Gian Antonio Susto, Angelo Cenedese, and Matteo Terzi. Chapter 9 - time-series classification methods: Review and applications to power systems data. In Reza Arghandeh and Yuxun Zhou, editors, *Big Data Application in Power Systems*, pages 179–220. Elsevier, 2018.

[96] Thomas L. Szabo. 5 - Transducers. In Thomas L. Szabo, editor, *Diagnostic Ultrasound Imaging*, Biomedical Engineering, pages 97–135. Academic Press, Burlington, 2004.

[97] Aleem Dad Khan Tareen, Malik Sajjad Ahmed Nadeem, Kimberlee Jane Kearfott, Kamran Abbas, Muhammad Asim Khawaja, and Muhammad Rafique. Descriptive analysis and earthquake prediction using boxplot interpretation of soil radon time series data. *Applied Radiation and Isotopes*, 154:108861, 2019.

[98] Marek Svepes Zdenek Rosa Hana Kubatova Tomas Cejka, Vaclav Bartos. Nemea: A framework for network traffic analysis. In *12th International Conference on Network and Service Management (CNSM 2016)*, 2016.

[99] Yuerong Tong, Jingyi Liu, Lina Yu, Liping Zhang, Linjun Sun, Weijun Li, Xin Ning, Jian Xu, Hong Qin, and Qiang Cai. Technology investigation on time series classification and prediction. *PeerJ Computer Science*, 8:e982, 2022.

[100] Jacob T. VanderPlas. Understanding the lomb–scargle periodogram. *The Astrophysical Journal Supplement Series*, 236(1):16, may 2018.

[101] Aditya Vikram and Mohana. Anomaly detection in network traffic using unsupervised machine learning approach. In *2020 5th International Conference on Communication and Electronics Systems (ICCES)*, pages 476–479, 2020.

[102] Ly Vu et al. Time series analysis for encrypted traffic classification: A deep learning approach. In *2018 18th International Symposium on Communications and Information Technologies (ISCIT)*. IEEE, 2018.

[103] William Wei. *Time Series Analysis: Univariate and Multivariate Methods*, volume 33. 01 1989.

[104] Qingsong Wen, Kai He, Liang Sun, Yingying Zhang, Min Ke, and Huan Xu. Robustperiod: Robust time-frequency mining for multiple periodicity detection. In *Proceedings of the 2021 International Conference on Management of Data*, pages 2328–2337, 2021.

[105] Seth J Worley et al. Comparison of Time Domain and Frequency Domain Variables From the Signal-averaged Electrocardiogram: A Multivariable Analysis. *Journal of the American College of Cardiology*, 11(5), 1988.

[106] Zhengzheng Xing, Jian Pei, and Philip S. Yu. Early classification on time series. *Knowledge and Information Systems*, 31(1):105–127, 2012.

[107] Hao Yin et al. Network traffic prediction based on a new time series model. *International Journal of Communication Systems*, 2005.

[108] Scott L. Zeger, Rafael Irizarry, and Roger D. Peng. On time series analysis of public health and biomedical data. *Annual Review of Public Health*, 27(1):57–79, 2006. PMID: 16533109.

[109] Hui Zeng, Zhiying Peng, XiaoHui Huang, Yixue Yang, and Rong Hu. Deep spatio-temporal neural network based on interactive attention for traffic flow prediction. *Applied Intelligence*, pages 1–12, 2022.

[110] Yi Zheng, Qi Liu, Enhong Chen, Yong Ge, and J Leon Zhao. Time series classification using multi-channels deep convolutional neural networks. In *International conference on web-age information management*, pages 298–310. Springer, 2014.

Publications of the Author

Reviewed Relevant Publications of the Author

[A.1] Koumar Josef, Hynek Karel, Čejka Tomáš. "NetTiSA: Extended IP Flow with Time-series Features for Universal Bandwidth-constrained High-speed Network Traffic Classification" The International Journal of Computer and Telecommunications Networking (COMNET), January 2024.

[A.2] Koumar Josef, Hynek Karel, Čejka Tomáš. "Network Traffic Classification based on Single Flow Time Series Analysis" 2023 19th International Conference on Network and Service Management (CNSM). IEEE, 2023.

[A.3] Koumar Josef, Plný Richard, Čejka Tomáš. "Enhancing DeCrypto: Finding Cryptocurrency Miners based on Periodic Behavior" 2023 19th International Conference on Network and Service Management (CNSM). IEEE, 2023.

[A.4] Koumar Josef, and Čejka Tomáš. "Network traffic classification based on periodic behavior detection." 2022 18th International Conference on Network and Service Management (CNSM). IEEE, Thessaloniki, Greece, 2022.

[A.5] Koumar Josef, Čejka Tomáš, "Unevenly spaced time series from network traffic" In: Proceedings of the 7th edition of the Network Traffic Measurement and Analysis Conference (TMA) 2023.

Other Reviewed Publications of the Author

[A.6] Pešek Jaroslav, Plný Richard, Koumar Josef, Jeřábek Kamil, Čejka Tomáš, "Augmenting monitoring infrastructure for dynamic software-defined networks" In: Proceedings of SpliTech 2023: International Conference on Smart and Sustainable Technologies 2023.

[A.7] Lukáš Jančička, Koumar Josef, Dominik Soukup, Čejka Tomáš, "Analysis of Statistical Distribution Changes of Input Features in Network Traffic Classification Domain" In: Proceedings of NOMS 2024 (will be published).

Published datasets of the Author

[A.8] Josef Koumar, Tomáš Čejka. (2023). "CESNET-USTS23: a benchmark dataset of Unevenly spaced time series from network traffic" [Data set]. Zenodo. https://doi.org/10.5281/zenodo.7923745

[A.9] Josef Koumar, Richard Plný, Tomáš Čejka. (2023). "CESNET-MINER22-TS: Periodic Behavior Features of Cryptomining Communication" [Data set]. Zenodo. https://doi.org/10.5281/zenodo.8033351

[A.10] Josef Koumar, Karel Hynek, Tomáš Čejka. (2023). "Network traffic datasets created by Single Flow Time Series Analysis" [Data set]. Zenodo. https://doi.org/10.5281/zenodo.8035724

[A.11] Josef Koumar, Karel Hynek, Jaroslav Pešek, Tomáš Čejka. (2023). "Network traffic datasets with novel extended IP flow called NetTiSA flow" [Data set]. Zenodo. https://doi.org/10.5281/zenodo.8301043

Work on Projects

[A.12] member of team of solvers, ongoing project: SmartADS TAČR EPSILON project, CESNET, a.l.e

[A.13] member of team of solvers, ongoing project: CYBERTHREATS - Use of artificial intelligence for defence against cyber security attacks (OYCESNET20221), CESNET, a.l.e

[A.14] member of team of solvers, ongoing project: Flow-based Encrypted Traffic Analysis (VJ02010024), Faculty of Information Technology, Czech Technical University in Prague

[A.15] member of team of solvers, ongoing project: SGS20/210/OHK3/3T/18, Czech Technical University in Prague

[A.16] member of team of solvers, ongoing project: SGS23/207/OHK3/3T/18, Czech Technical University in Prague

Organizing Conferences

[A.17] Posters Co-Chair at The 20th International Conference on Network and Service Management (CNSM), 2024

[A.18] Student poster session Co-Chair at The 12th Prague Embedded Systems Workshop (PESW), 2024