mathematics

MDPI

Article

Machine Recognition of DDoS Attacks Using Statistical Parameters

Juraj Smiesko, Pavel Segec and Martin Kontsek *

Department of InfoComm Networks, Faculty of Management Science and Informatics, University of Zilina, 010 26 Zilina, Slovakia
* Correspondence: martin.kontsek@fri.uniza.sk

Abstract: As part of the research in the recently ended project SANET II, we were trying to create a new machine-learning system without a teacher. This system was designed to recognize DDoS attacks in real time, based on adaptation to real-time arbitrary traffic and with the ability to be embedded into the hardware implementation of network probes. The reason for considering this goal was our hands-on experience with the high-speed SANET network, which interconnects Slovak universities and high schools and also provides a connection to the Internet. Similar to any other public-facing infrastructure, it is often the target of DDoS attacks. In this article, we are extending our previous research, mainly by dealing with the use of various statistical parameters for DDoS attack detection. We tested the coefficients of Variation, Kurtosis, Skewness, Autoregression, Correlation, Hurst exponent, and Kullback–Leibler Divergence estimates on traffic captures of different types of DDoS attacks. For early machine recognition of the attack, we have proposed several detection functions that use the response of the investigated statistical parameters to the start of a DDoS attack. The proposed detection methods are easily implementable for monitoring actual IP traffic.

Keywords: IP traffic description; DDoS attack detection; coefficient of variation; kurtosis; skewness; autoregression; correlation; hurst exponent; Kullback–Leibler divergence; predicting tunnels; recognition of DDoS attacks

MSC: 62P99

Citation: Smiesko, J.; Segec, P.; Kontsek, M. Machine Recognition of DDoS Attacks Using Statistical Parameters. *Mathematics* **2024**, *12*, 142. https://doi.org/10.3390/math12010142

Academic Editor: Cheng-Chi Lee

Received: 20 November 2023
Revised: 14 December 2023
Accepted: 22 December 2023
Published: 31 December 2023

1. Introduction

Nowadays, we use technology in our everyday lives, and it helps simplify many mundane or even more advanced tasks. Not many of us think about the possibility of losing access to the technology advancements and how it will be hard for us to adjust to living without it. The threats to most of the tools are, however, very real, and they are attacked almost constantly, even without us knowing, which is a significant cybercrime issue. Fortunately, most nations worldwide are investing heavily in building Security Operation Centers to help with the awareness of cyber attacks and their prevention. Both universities and the majority of high schools in Slovakia are connected to the Internet through the SANET network infrastructure, which, as a public-facing network, is the target of a huge amount of attacks ranging from simple bruteforce login tryouts up to more sophisticated Distributed Denial of Service (DDoS) attacks.

DDoS could be considered an improved version of simpler DoS attacks. In both forms, the attacker tries to make the service provided to clients by the server inoperable or otherwise impaired. This means that the responses by the server could be much slower or even missing compared to the unaffected server. Unlike DoS attacks, which originate in one place only, a DDoS attack consists of a multitude of originating sources, which are often called bots. Bots are usually network-connected devices that are controlled by the attacker from a single point, also called the Command and Control server.

There are three main types of DDoS attacks mentioned in [1]:

- Application attacks, exploiting some well-known or even unknown vulnerabilities, also called zero-day attacks, in application protocol or service, are the first and most serious type. The attackers utilizing this type of attack are very effective because even with a small number of controlled devices, they can cause critical service outages. Protecting against them is intricate because of the difficult detection and mitigation by administrators.
- Protocol attacks affecting transport protocols, such as TCP or UDP, could be considered a second type. Creating a vast amount of TCP connections is very effective and can easily lead to possible connection limit exhaustion. This type of attack is found not only on routers and firewalls, but it can also affect servers or load balancers, which are trying to distribute the load between multiple servers.
- Volume-based attacks, being the last but not least important type, are the simplest of the three types. The attacker is trying to exhaust the server's available bandwidth to cause network congestion. As a result of this attack type, the server can neither get the request from the client nor respond to a received request.

However, the latest attacks could not be classified into the mentioned types because they combine several types of characteristics. Very often, DDoS attacks provide cover for sophisticated malware injections by attackers, which is even harder for security analysts or even automated tools to detect. To make things even worse, they can be used to ex-filtrate classified information from the infected devices, which later become part of the botnet themselves and are often referred to as "zombie devices" [2]. Some cybercrime groups or individuals even sell access to large botnets for staggering prices, as creating ones without sophisticated attack techniques is hard.

Defense against DDoS attacks, according to [3], includes three main parts: monitoring, detection, and response. The monitoring phase plays a key role in obtaining information about the network services the user provides. Detection methods are built on data collected during monitoring, and network patterns and anomalies or incidents are analyzed. The response phase is triggered after an attack is identified through detection methods. This includes implementing firewall rules as the first line of defense, detecting the threat, and immediately notifying the network security team.

Our university's Internet connection through the academic high-speed network, called the Slovak Academic Network (SANET), is subject to constant cyber threats. Devices connected to this network face a variety of attacks, including simpler brute force attacks on SSH, RDP, or HTTP/HTTPS, to more serious DDoS attacks. Network protection is financially demanding, as increasingly powerful network equipment is required to withstand more sophisticated attacks, especially as SANET currently consists of several 100 Gb links on one segment. That is why, as part of the SANET II project "Research in the SANET network and possibilities for its further use and development", we were trying to find computationally efficient statistical methods and create a machine learning system to detect DDoS attacks in real time. The method was designed in a way that allowed easy implementation of these methods into a hardware probe to monitor IP traffic in real time on any connected network segment.

The primary objective of the SANET II project is to implement research findings through innovative services and technologies within the distribution network, prioritizing security and dependability. The project's suggested methods and principles aim to expedite the adoption of technologies, ensuring a more effective and secure transmission of specific data. The objective is to devise fresh models and distribution approaches, anticipating future interdisciplinary adjustments. Progress in the development of network infrastructure in this realm will not only enhance the ability of the scientific and research community to distribute, store, and exchange R&D data efficiently but will also pave the way for potential adaptations in the realm of Industry 4.0. This involves modifying the proposed concepts for machine communication mechanisms within a vast network environment. Our team is actively engaged in advanced flow monitoring and assessing security events for both networks and Cloud Computing systems. Numerous articles authored by our team

delve into CC systems [4], their security architecture [5], the management of cybersecurity incidents, and the establishment of a packet capture infrastructure to generate valuable datasets [6,7].

At the beginning of a DDoS attack, there is a significant increase in the peak rate and average rate of IP traffic. Detecting an attack using these rates is insufficient. A significant peak can occur randomly, even during the standard traffic, and the moving average rate, which is calculated in a time window and increases linearly during a DDoS attack. Even with the so-called Low Rate DDoS attacks, this increase is insignificant. The mentioned factors can cause the detection of false positives or late recognition of a DDoS attack. The peak rate and average rate monitoring do not allow the recognition of a change in the probabilistic character of the monitored flow, which occurs during an attack due to generating a number of fraudulent packets.

In our research, we try to find such probabilistic characteristics of the IP flow that, by significantly changing their values, would be able to react to the start of a DDoS attack in time. At the same time, we are trying to create prediction methods that would create intervals of permissible values for the considered characteristics while monitoring normal network traffic. Exceeding the interval limits at the time the DDoS attack begins allows us to detect the attack early. This approach presupposes the input of normal IP traffic at the beginning of monitoring but does not require previous "learning" of already recorded attacks for detection. We recommend the research presented in this article on machine learning methods without a teacher.

The article continues and extends the work "One-Parameter Statistical Methods to Recognize DDoS Attacks" [8]. In the third chapter, we describe the processing of IP traffic and present the eight measured DDoS attacks we used. In the fourth chapter, we analyze the reaction of various statistical coefficients to the start of attacks in the traffic flow. These are, in order, coefficient of variation, kurtosis, skewness, entropy, Hurst exponent, autoregression coefficient, correlation coefficient, and Kullback–Leibler Divergence. In the fifth chapter, we deal with various prediction methods, and in the last chapter, we propose several detection functions designed for fast machine recognition of DDoS attacks. Finally, we summarized the results, recommendations, and suggestions for further research direction in the results and discussion.

2. Related Work

Many mathematical methods try to detect a DDoS attack. Among the main ideas is the monitoring of changes in the probability distribution of the occurrence of packets during the transition from standard traffic to attack. For this reason, the statistical moments describing the distribution of packet occurrences will change, for example, average rate, variance, spiciness coefficient [9], measures of periodicity, kurtosis, skewness, and self-similarity [10,11].

More complicated mathematical methods include regression and autocorrelation analysis. Using multiple regression analysis, the strength of a DDoS attack is estimated [12]. Using a regression model for predicting the number of zombies in DDoS attacks is discussed in [13]. In [14], they use a change of autocorrelation coefficients to detect an attack. Autocorrelation in the convolution of legitimate and attack traffic (cross-correlation method) is discussed in [15].

Another important statistical characteristic used is the Hurst exponent, which describes the self-similarity of time flow. The change in self-similarity occurs during the transition from normal traffic to one containing attacks. The Hust exponent was used to identify a DDoS attack in [16–18]. A comparison of average Hurst exponent values between standard and attacking traffic can be found in [19]. An article [20] deals with the combination of correlation and the Hurst exponent. The authors of the article [21] used an autoregressive system for estimating the variance of the Hurst coefficient to detect changes in the flow. The use of self-similarity and Renyi entropy can be found in [22]. Fractal

analysis is closely related to self-similarity; its application to detect attacks can be found in [23]. Authors in [24] deal with a combination of fractal and recurrent functions.

Good results are achieved by Machine Learning (ML) with the use of Neural networks [25]. The authors in [26] used GAN networks for detection. The combination of Autoencoders and Deep Convolution GAN Networks for determining anomalies in IP flow is discussed in [27]. The GAN Network with two Discriminators is used in [26,28].

An effective method for detection is also Principal Component Analysis (PCA). In [29], PCA is used to indicate anomalies. In [30], PCA is used for dimensionality reduction of the IP flow dataset attributes.

Low-rate DDoS attacks form a special category of attacks. With this type of attack, there is no significant increase in the moving average rate. In [31], they used several metrics and entropies for low-rate attack detection. Authors in [32] recognize entropy attacks using the difference in packet size between normal and attacking traffic. Self-similarity has been used in [33], and Queue Management Algorithms (RED and REM) have been used against low-rate DDoS attacks in [34].

Other methods include detecting attacks using wavelets (wavelet analysis) [35–37], blockchain [38,39], genetic algorithms and random forest [40], and the use of various spectral and cluster analyses and mathematical models of the SDN networks is mentioned in [41].

Most mathematical methods used for machine learning to detect DDoS attacks use standard datasets, e.g., MIT outside of normal traffic, CAIDA-2007 DDoS attacks, TUDDoS dataset, etc. Based on the patterns in the datasets, areas describing standard and offensive traffic are created for machine detection. These areas can contain, for example, simple samples, corresponding values of statistical parameters, or other characteristics. The unknown sample is then recognized based on these previously "learned" areas.

Unlike the previous methods mentioned in this chapter, we try to find statistical characteristics that, during online monitoring of IP traffic, would quickly react to the beginning of a DDoS attack by significantly changing their values. The next step is determining predictive methods and detection functions, allowing the machine to recognize these changes during attacks. Such recognition is one of the methods of machine learning without a teacher.

3. Processing of IP Records

3.1. Ip Flow Description

We can describe the packet flow in several ways, depending on which mathematical model of the IP network we want to use. In Queuing Theory, the oldest model used is the Jackson Network [42,43]. All input and output flows to nodes are modeled using a Poisson process. Another stochastic model that uses the Large Deviation Principle describes flows using their Effective Bandwidths [44,45]. In Network Calculus, which represents a deterministic model of an IP network, input–output flows are bounded by subadditive curves [46].

In both deterministic and stochastic models of the IP network, the cumulative process $A(s,t)$ is used to analyze the IP flow, which describes the occurrence of packets in the time interval $\langle s,t\rangle$. For models with discrete time, there are certain time slots [ts] given for analysis or sample windows in which the number of occurring packets a_i (increments in the time i) are recorded:

$$\forall t,s \in R^+; \quad A(s,t) = A(0,t) - A(0,s) = \sum_{i=s}^{t} a_i \tag{1}$$

In the stochastic model, $A(s,t)$ is a cumulative random chain, $A(s,t,\omega)$ and a_i represent some non-negative discrete random variables $a_i(\omega)$. Assuming the flow's stationarity, the cumulative chain's designation is simplified to $A(s,t,\omega)$, and the random variables $a_i(\omega)$ have the same probability distribution.

In the case of measuring IP traffic using W-shrike, we have at our disposal a vector of the cumulative time of packet arrivals T_j. After choosing the size of the time slot and using addition, we obtain the values of the increments of the IP flow in the given time slot, see Figure 1:

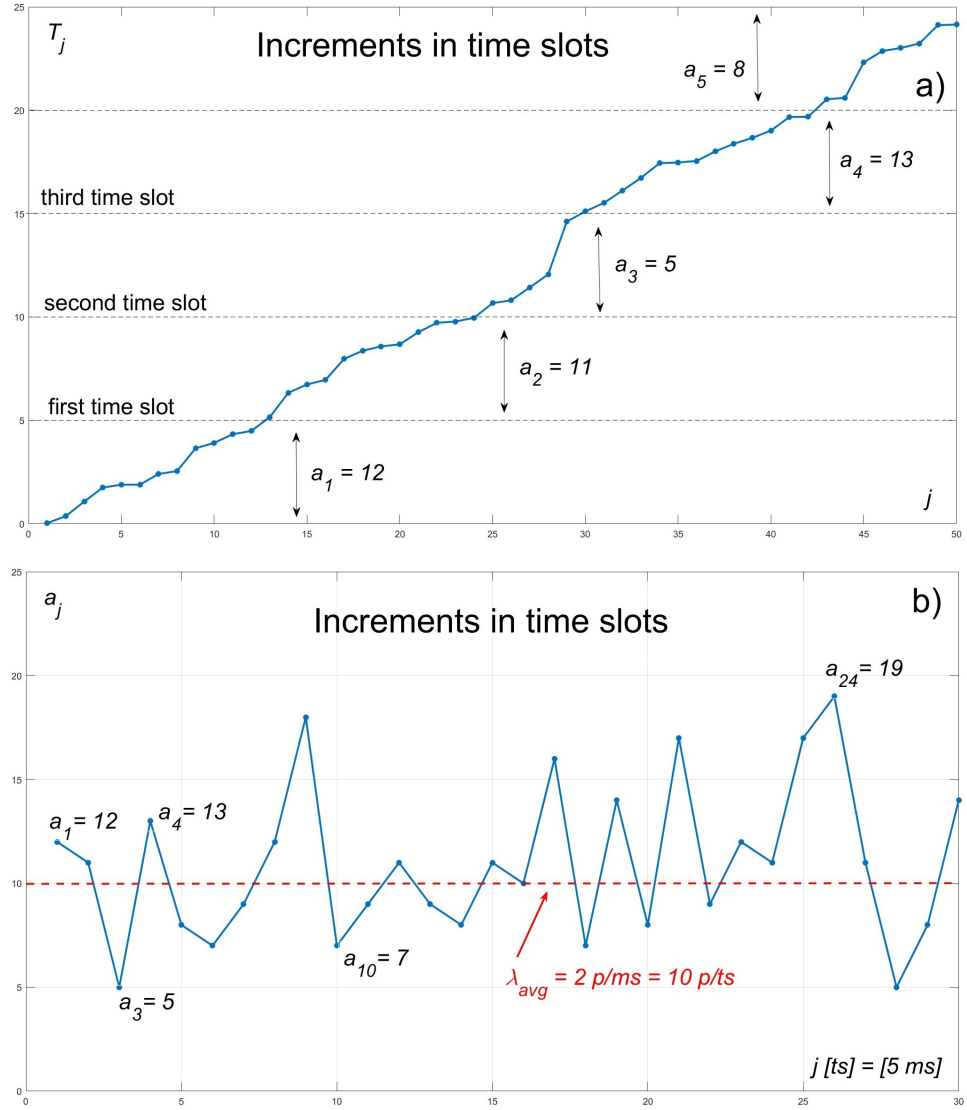

Figure 1. Description of IP flow: (**a**) cumulative time, (**b**) increments in time.

We used a Poisson flow simulation with an average rate $\lambda_{avg} = 2p/ms$ to demonstrate the method of sampling the time record of the measured traffic from Wireshark into the increments of flow time series. In Figure 1a, we have shown 50 cumulative exponential values of variables T_j, representing the occurrence of packets in time. For the size of the time slot or sample window, we chose $ts = 5ms$. After addition, we obtained increments a_i with Poisson distribution with average rate $\lambda_{avg} = 10p/ts$, $a_i(\omega) \sim Po(10)$. We displayed the first 30 increment values as a time series in Figure 1b.

In our article, we want to use statistical coefficients to describe the IP flow. Therefore, we do not use the description of the flow using the cumulative stochastic process $A(s, t, \omega)$ with random increments $a_i(\omega)$, but it is sufficient for us to use a significantly simpler model. We will consider the vector of sampled increments $\mathbf{a} = (a_1, \ldots, a_N)$ to be the N realization of some random variable $X(\omega)$, which we will use to estimate the value of $\widehat{\theta}$ of some statistical parameter $\theta(\omega)$. We denote these N realizations as compute window $cw = \mathbf{a}$ of size $|cw| = N$. To detect an anomaly in the IP flow, we must create a time series of values

of some statistical coefficient, which we gradually calculate from mutually overlapping time windows (the so-called sliding coefficient). We chose the overlap by one time slot or sample window for early anomaly detection. The calculation of two successive values of the estimate $\widehat{\theta}$ is obtained from two consecutive overlapped computed windows:

$$\widehat{\theta}_1 = \widehat{\theta}(\mathbf{a}_1) = \widehat{\theta}(a_1, \ldots, a_N), \qquad \widehat{\theta}_2 = \widehat{\theta}(\mathbf{a}_2) = \widehat{\theta}(a_2, \ldots, a_{N+1}) \tag{2}$$

Statistical coefficients whose values are always obtained only from one compute window are called *One-window parameters*. These are actually estimates of the probabilistic characteristics of $X(\omega)$.

For the following demonstration of computing windows, we used a capture of a measured DDoS attack, Figure 2:

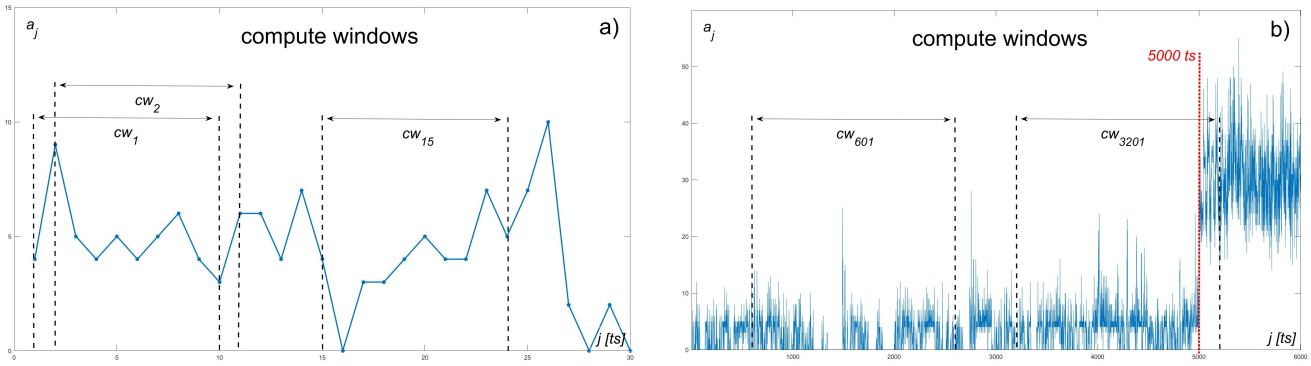

Figure 2. Compute windows: (**a**) $|cw| = 10\ ts$, (**b**) $|cw| = 2000\ ts$.

In addition to one-window parameters, we also deal with estimates of the probability characteristics of two random variables $X(\omega)$ and $Y(\omega)$ (for example, coefficient of correlation). For the values of such a parameter, we need two computed windows. We call such characteristics *two-windows parameters*. We will consider the vector increments $(a_1, \ldots, a_N)$ as the N realization of $X(\omega)$ and the vector $(a_2, \ldots, a_N)$ as the N realization of $Y(\omega)$. After calculating the value of the coefficient, we shift the entire pair in time by 1ts, as in one-window parameters, Figure 3:

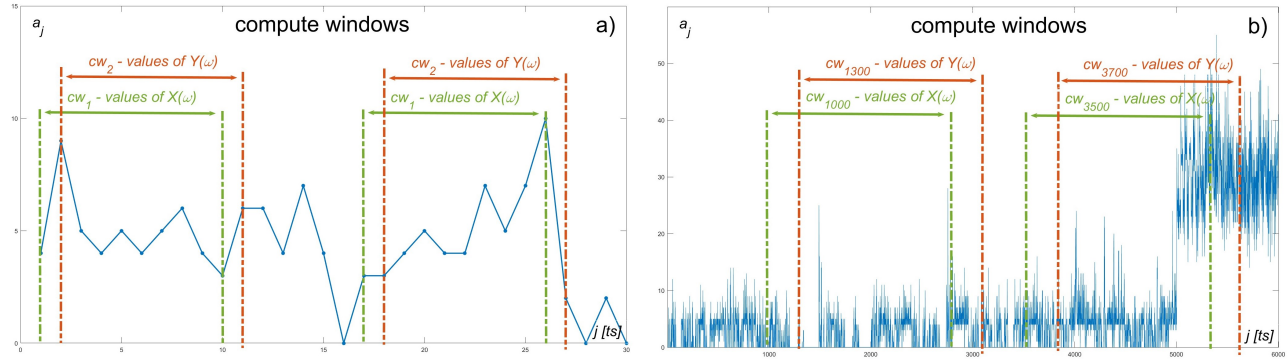

Figure 3. Compute windows pair: (**a**) $|cw| = 10\ ts$, shift 1 ts, (**b**) $|cw| = 2800\ ts$, shift 300 ts.

By gradually moving one compute window or pairs of windows, we obtain a time series of the estimated values of the given statistical parameter. When using a shift of one-time slot and sizes, e.g., $|cw| = 100\ ts$ at the start of the attack, we calculate the parameter value from the computed window, which contains 1% of the attack traffic. Our effort is to find statistical parameters that react relatively quickly to the occurrence of offensive traffic. By successively moving the computed window, we obtain a time series of estimated values of the given statistical parameter.

3.2. Types of DDoS Attack

In this chapter, we present selected captures of real DDoS attacks, on which we present the course of the values of individual monitored statistical coefficients. We intentionally omitted several experiments that we performed on various simulated scenarios where we used IP flow generated using 2-state On/Off processes, using Poisson and Pareto distributions, and also using the MNIST and CIFAR databases. In these simulated scenarios, the selected statistical parameters worked "exceptionally well", and we acquired an initial idea of the effectiveness of the use of individual parameters in recognizing changes and increases in simulated traffic. However, the situation changed significantly when deployed to detect a real attack. We will analyze the response of statistical coefficients on eight selected attacks.

We obtained captures from the following datasets, e.g., ISCX 2012 and CIC-IDA 2017, but mainly from our own custom dataset [47]. More detailed information about the attacks is available in [8,48,49].

We divided the attack captures into several types according to the course of the observed statistical coefficients (see the next chapter). The first type represents standard attacks (N-normal) in which the coefficients reacted similarly to simulated scenarios [8], Figures 4 and 5:

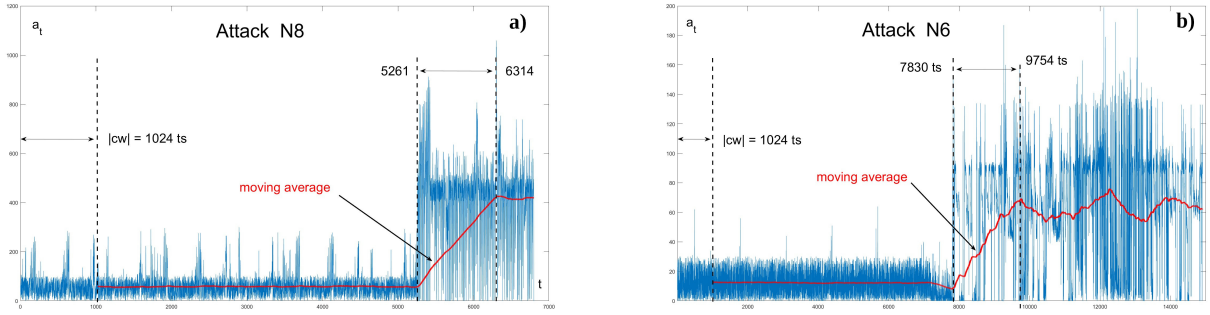

Figure 4. Increments of normal Distributed Denial of Service (DDoS) attack, (**a**) N8, (**b**) N6.

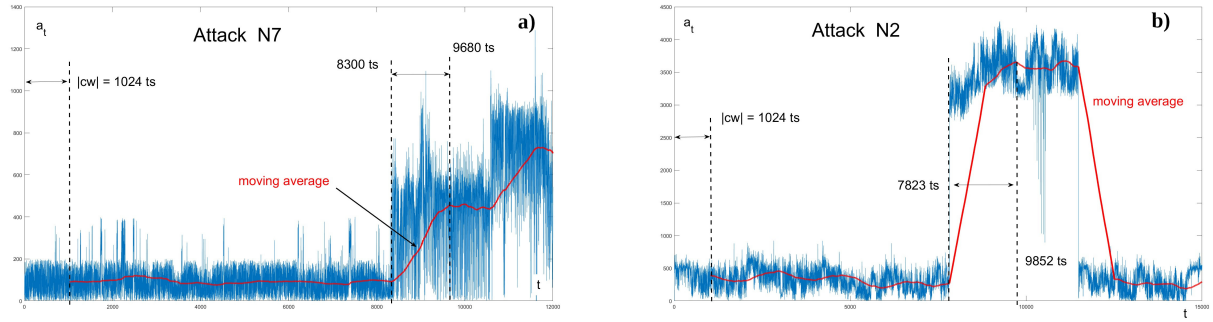

Figure 5. Increments of normal Distributed Denial of Service (DDoS) attack, (**a**) N7 and (**b**) N2.

The graphs show the flow increments a_t, moving average $m(t)$ with $|cw| = 1024\ ts$, and time intervals from the beginning of the attack to the moment when the average reached its local maximum. We will later use these intervals to evaluate other statistical parameters' effectiveness objectively.

The other two captures, Figure 6, at first glance, also represent standard attacks; the standard traffic has a stationary character, and the offensive traffic has several times the average rate. However, the values of the coefficients behaved differently than with normal attacks, which is why we labeled them special attacks (S-special).

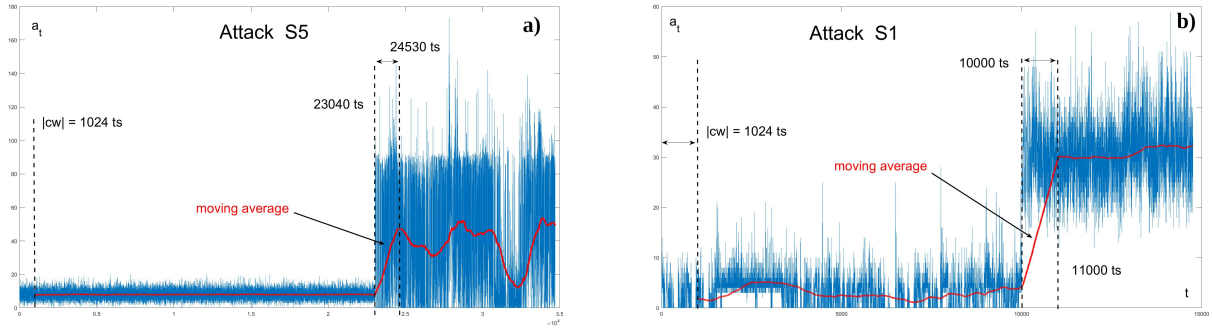

Figure 6. Increments of special Distributed Denial of Service (DDoS) attack, (**a**) S5 and (**b**) S1.

The seventh capture in the sequence, Figure 7a, is a representative of the so-called Low Rate (LR) DDoS attacks, i.e., attacks with a low average rate of flow [50,51]. As the last record, we mention a problematic attack, Figure 7b, in which the monitored coefficients mostly failed to capture the starting point of the offensive traffic (P-problem). It was a DDoS attack captured using the core server at the University of Žilina [8]. According to the course of the moving average, it also belongs to the low-rate attack. We devoted a special section to this attack.

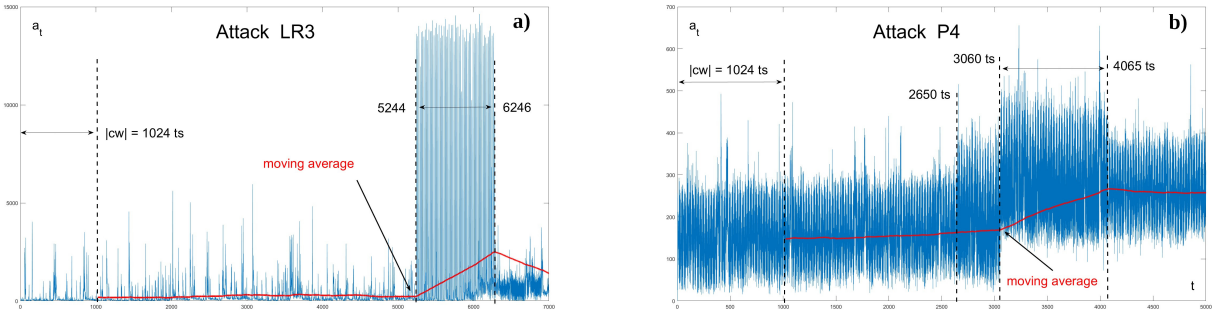

Figure 7. Increments of (**a**) Low-rate Distributed Denial of Service (DDoS) attack LR3 and (**b**) Problem attack P4.

For the gradual calculation of coefficient values, we used a shift of the computed window by a one-time slot. When conducting experiments with records of DDoS attacks, we used different powers of two (256, 512, 1024, 2048, 4096) to estimate the Hurst exponent for computing window sizes. After a subjective evaluation of the course of the values of the investigated coefficients, we decided to consider only further compute windows of size $|cw| = 1024\ ts$. At $|cw| = 512\ ts$, there was a significant "ripple" in the courses of some parameters; at $|cw| = 2048\ ts$, the time for calculating the parameters, especially the Hurst exponent, increased significantly.

We can observe the linear growth of the moving average on all the listed records. Of course, this growth depends on the size of the compute window. For the sake of comparison, we used the same size $|cw| = 1024\ ts$ and shifted it by $1\ ts$. This size guarantees that the average does not react to random peaks in the flow.

We assume that in a DDoS attack, the overall probabilistic structure of the flow will change due to the generation of a large number of flood packets. Our effort is to determine such statistical parameters that react to a DDoS attack significantly faster and more pronounced than the linear growth of the moving average.

4. Responses of Statistical Parameters to a DDoS Attack

4.1. One-Window Parameters

4.1.1. Coefficient of Variation

The basic probabilistic characteristics of the given random variable $X(\omega)$ are the first initial moment μ (mean) and the second central moment σ^2 (variance). Using their mutual quotient, the *coefficient of variation* v is defined. The coefficient of variation of the random variable $X(\omega)$ represents the ratio between the standard deviation and the mean value [52].

$$\mu = EX(\omega), \qquad \sigma^2 = DX = E[X(\omega) - \mu]^2, \qquad V = \frac{\sigma}{\mu} \tag{3}$$

Statistical estimates of the above characteristics will be denoted as average rate λ_{avg}, sample variance S^2, and sample coefficient of variation V:

$$\lambda_{avg} = \frac{1}{N}\sum_{i=1}^{N} a_i, \qquad S^2 = \frac{1}{N-1}\sum_{i=1}^{N}(a_i - \lambda_{avg})^2, \qquad V = \frac{S}{\lambda_{avg}} \tag{4}$$

When calculating coefficient values using overlapped compute windows, we will talk about moving coefficients, for example, moving average and moving sample variation, and denote them as $m(t)$ and $V(t)$.

In the case of standard DDoS attacks (type N), the standard traffic has the character of a stationary flow, whereby the standard deviation σ acquires significantly smaller values compared with the average rate. At the start of a DDoS attack, the average rate will increase many times, and the standard deviation value will also increase, even faster than the linear trend. For this reason, at the moment of a DDoS attack's starting point, the variation coefficient exceeds the value $V(t) = 1$. This is how the coefficient reacted to all analyzed standard attacks of the N type and to an S5 attack, for example, N8, Figure 8: Please check all figures.

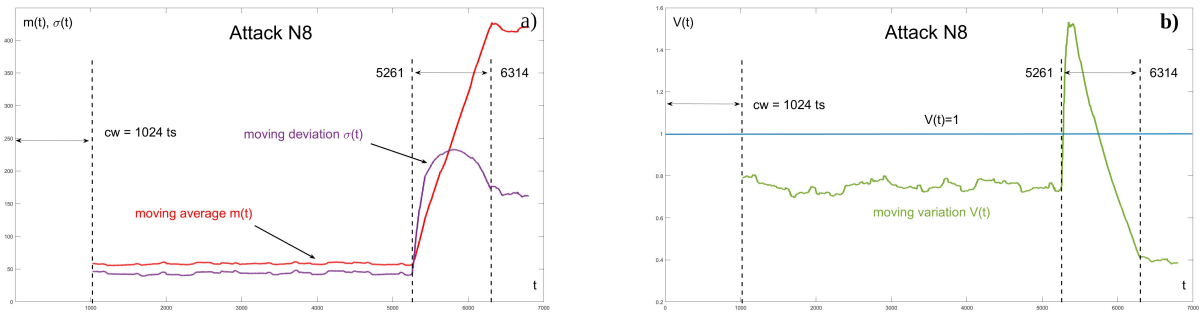

Figure 8. Attack N8. (**a**) Moving average and sample deviation, (**b**) moving sample variation.

During a non-standard attack of the S1 type, the coefficient $V(t)$ exceeded the value of 1 several times already during normal traffic, Figure S3. During the entire capture of low-rate attack LR3 $V(t) > 1$ held, Figure 9:

During the problematic attack, the P4 coefficient did not exceed the value of 1 at all. The progress of the other recordings is given in the article's appendix.

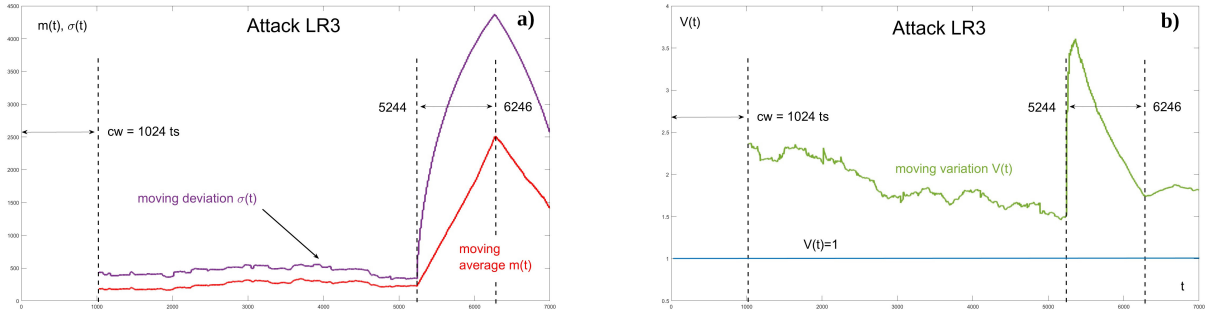

Figure 9. Attack LR3. (**a**) Moving average and sample deviation, (**b**) moving sample variation.

4.1.2. Kurtosis Coefficient and Skewness Coefficient

The other basic characteristics of the random variable $X(\omega)$ are *kurtosis coefficient* and *skewness coefficient*. It is actually the third and fourth central moment scaled standard deviation:

$$\mu_3 = \frac{E[X(\omega) - EX(\omega)]^3}{\sigma^3}, \qquad \mu_4 = \frac{E[X(\omega) - EX(\omega)]^4}{\sigma^4} \tag{5}$$

Both coefficients certainly describe the properties of the probability distribution of the random variable $X(\omega)$. The kurtosis coefficient is a measure of the asymmetry of the probability distribution around the mean value of the random variable. The skewness coefficient describes how much the peak of the curve of the density function differs from the Gaussian density function.

In our case of realization of the variable $X(\omega)$, the values of the increments are a_i, and the coefficients describe the properties of the probability distribution of the increments in the given calculation window *cw*. We denote their estimates as K and S_{kw}. We get the first estimate values from the compute window $(a_1, \ldots, a_N)$:

$$K = \frac{\frac{1}{N-1}\sum_{i=1}^{N}\left(a_i - \lambda_{avg}\right)^3}{\left[\frac{1}{N-1}\sum_{i=1}^{N}\left(a_i - \lambda_{avg}\right)^2\right]^{3/2}}, \qquad S_{kw} = \frac{\frac{1}{N-1}\sum_{i=1}^{N}\left(a_i - \lambda_{avg}\right)^4}{\left[\frac{1}{N-1}\sum_{i=1}^{N}\left(a_i - \lambda_{avg}\right)^2\right]^2} \tag{6}$$

When processing the attack traffic into the compute window, we assumed a significant change in the probability distribution and, thus, also a change in the coefficients. The assumption was confirmed for almost all analyzed flows, for example, N8, Figure 10:

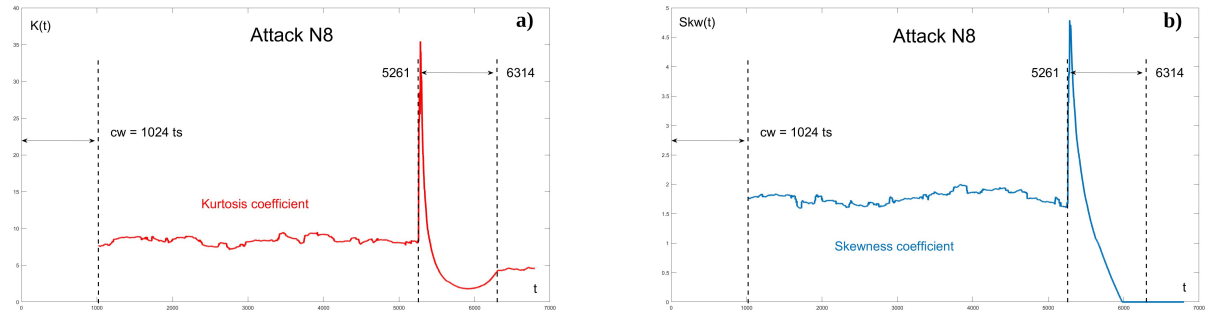

Figure 10. Attack N8. (**a**) Moving kurtosis coefficient. (**b**) Moving skewness coefficient.

Both statistically computed calculations reacted with a significant increase in their values right at the beginning of the DDoS attack. A similar situation occurred during the processing of recording S1 and low-rate attack LR3. We noticed a different behavior of the

coefficients only in the S1 attack, Figure 11. To visualize the reaction of the coefficients to the change in the nature of the traffic and to the occurrence of peaks, we displayed the values of the coefficients in a common graph together with the flow increments S1, Figure 11:

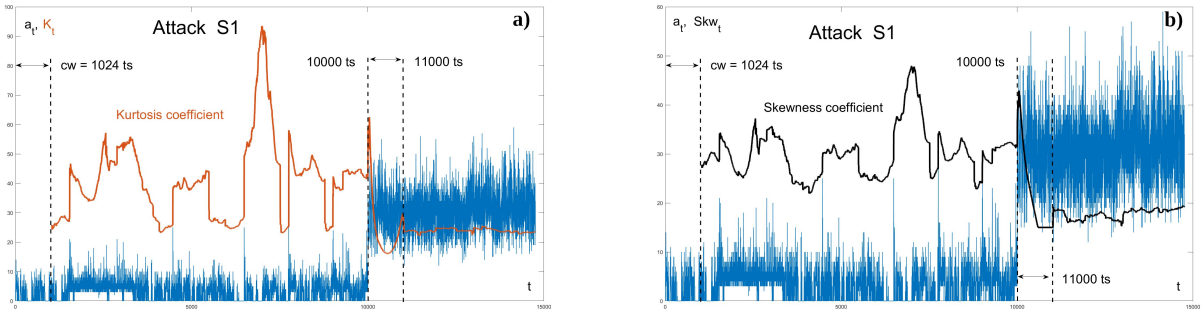

Figure 11. Attack S1. (**a**) Moving kurtosis coefficient. (**b**) Moving skewness coefficient.

Due to the non-stationarity of normal traffic and the frequent occurrence of high peaks in the S1 record, the values of both coefficients fluctuate strongly, due to which the jump at the starting point of the attack is lost in the overall flow of the values of both coefficients.

The coefficients K and Skw are estimates of the third and fourth central moments of the stochastic variable $X(\omega)$, which is why the course of their values is very similar. In the majority of the analyzed captures, they reacted to the start of a DDoS attack with a several-fold increase (peak) of values compared with the previous course. We will use this fact in machine recognition using prediction methods. The coefficient of variability reacted similarly, but in its case, the exceeding of the value $V(t) > 1$ can be used to detect an attack in several captures.

4.1.3. Entropy

Entropy is associated with terms, such as thermodynamics, statistical mechanics, or information theory. This physics quantity expresses the degree of randomness or the uncertainty in which some random event or signal occurs. We can then also represent this degree of randomness as size information that the given signal can transmit. Entropy, as well as kurtosis and skewness describe, in a sense, the probability distribution of increments a_i in the currently processed compute window:

$$\Pr(a_i = k) = p_k, \quad k = 0, \ldots, m \tag{7}$$

Let the size of the computed window be $|cw| = N$, and n_k represents the number of values $a_i = k$ in the given window cw. We denote the Entropy for the given window as H and its estimate E:

$$H = \sum_{k=0}^{m} p_k \ln p_k, \qquad E = \sum_{k=0}^{m} \left[\frac{n_k}{N}\right] \ln\left[\frac{n_k}{N}\right], \qquad k = 0, \ldots, m \tag{8}$$

When the attack traffic is gradually loaded into the current CW, we assume a significant change in the probability distribution and, thus, also a change in the entropy value. For illustration, we present two extreme cases, attacks N8 and N6, Figure 12:

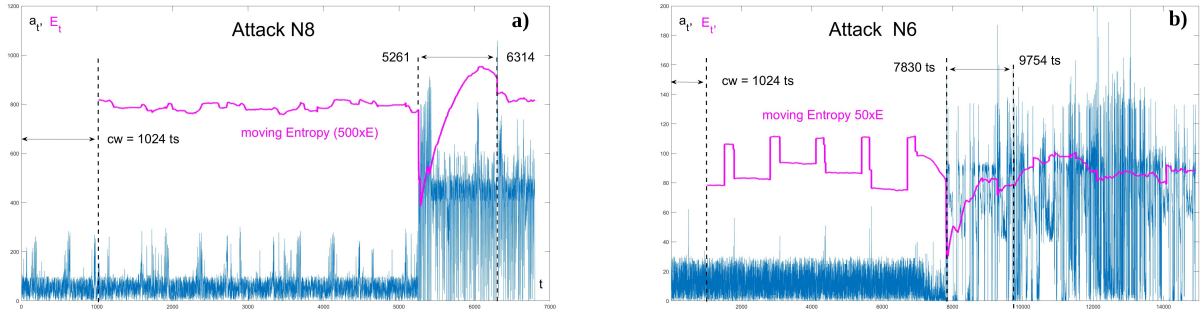

Figure 12. Increments and moving Entropy, (**a**) record N8, (**b**) record N6.

Entropy reacted with a significant drop in values at the beginning of the DDoS attack. In an ideal case, the decrease is significantly more pronounced than the previous course of Entropy. In the majority of recordings, however, the entropy values reacted significantly even to slight changes in the structure of standard traffic and to the occurrence of peaks in traffic flow.

Based on most of the experiments on various DDoS attacks, we concluded that Entropy is useless for attack detection; prediction of its progress would lead to the detection of many false attacks. For these reasons, we excluded Entropy from the investigated methods.

4.1.4. Hurst Exponent

Another examined characteristic was the Hurst exponent. The exponent expresses the degree of self-similarity of the time series. It is used in several areas of applied mathematics, including fractals and chaos theory, long-term memory processes, spectral analysis, and in sizing network parameters in Queueing theory.

When testing Hurst's reactions on simulated DDoS scenarios and on the first gathered real recordings, we got some interesting results [53]. The exponent had a tendency to "jump" to values close to H = 1 at the start of the attack. At the same time, during the processing of standard traffic, it remained in the range between 0.4 and 0.6, which corresponds to the values of a stationary random process. The possibility of using the Hurst exponent for machine recognition of an attack when a value close to H = 1 is exceeded [8], was drawn. When the experiments were carried out on other real captures of attacks, Hurst's exponent stopped responding ideally.

The Hurst coefficient cannot be calculated analytically; we can only estimate it statistically. Several methods are used to estimate the exponent, the main ones include the R/S statistic, the aggregated variance method, the absolute value method, the variance of the residuals, the Higuchi's method, the Modified Variance of Allan, the scale window variation, the Whittle estimator, etc. [54]. We used estimation using R/S Analysis [55] and Detrended Fluctuation Analysis (DFA). We do not mention individual estimation procedures due to their complexity; their description is given, for example, in [8,56,57]. Based on our empirical experience, we also used a modified estimate of the Hurst exponent using R/S Analysis, whereby we first removed their linear autoregressive trend from the incremental values in the given compute window, $x_t = a_t - (\alpha a_{t-1} + \beta)$, (mark RS AR(1)) [58].

In the following figures, we show two cases where, according to our subjective evaluation, the reaction from the point of view of machine recognition turned out to be very negative and absolutely ideal (other records are listed in the appendix). When processing recording LR3, Figure 13a, none of the Hurst exponent estimates reacted significantly to the onset of a DDoS attack. In the case of N8, Figure 13b, all three estimates reacted significantly, whereby the RS estimates exceeded the value of H = 1.

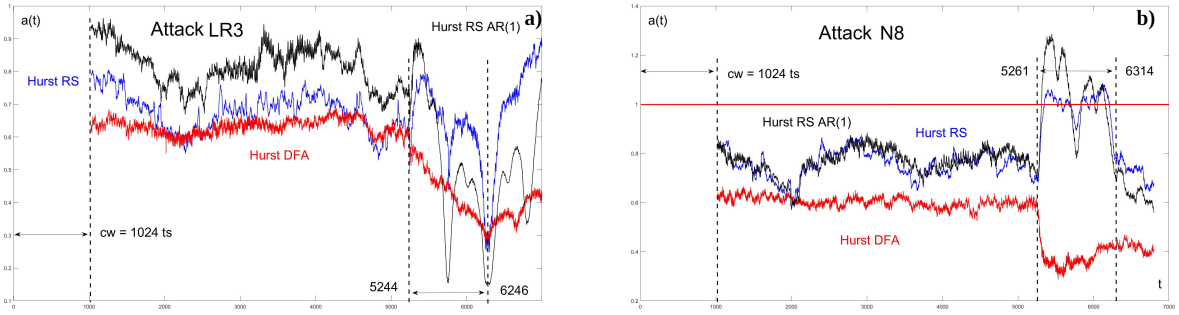

Figure 13. Hurst exponent. (**a**) Attack LR3. (**b**) Attack N8.

Overall, we can say that with DFA estimation, the values of the H-exponent at the moment of the attack mostly dropped significantly. In RS estimation, the course of values is often insignificant for machine recognition. However, after removing the linear autoregressive dependence, the H-exponent values increased significantly. In some cases, at moments of attack, the values of the exponent even exceeded the value of H = 1. However, this exceedance also occurred during standard traffic processing, for example, attack N2, Figure S17, and S1, Figure S19. Therefore, this property of the Hurst exponent can be used in machine recognition only in combination with other parameters.

4.2. Two-Windows Parameters

4.2.1. Autoregressive and Correlation Coefficients

Autoregressive and correlation coefficients are very similar probabilistic characteristics that express a certain kind of dependence between two random variables, in our case, between $X_t(\omega)$ and $X_{t-1}(\omega)$, i.e., variables whose realizations represent IP flow increments of a_i in the overlapped compute windows.

Correlation coefficient ϱ represents scaled covariance using standard deviations of individual random variables:

$$\varrho(X_t, X_{t-1}) = \frac{cov(X_t, X_{t-1})}{\sigma_{X_t}\sigma_{X_{t-1}}} = \frac{E[(X_t(\omega) - EX_t)(X_{t-1}(\omega) - EX_{t-1})]}{\sigma_{X_t}\sigma_{X_{t-1}}} \tag{9}$$

The autoregressive coefficient represents a linear dependence in an autoregressive model $AR(1)$, which assumes that the considered random process $\{X_t(\omega)\}_{t\in T}$ has the structure

$$X_t(\omega) = \beta X_{t-1}(\omega) + \varepsilon_t(\omega), \tag{10}$$

while random variables $\varepsilon_t(\omega)$ constitute white noise [59]. We denote the estimate of autoregressive coefficient β as c. For realizations of the random process $(a_1, a_2, \ldots, a_{N+1})$ holds

$$a_t = ca_{t-1} + \epsilon_t, \qquad t = 2, \ldots, N \quad \Rightarrow \quad \begin{bmatrix} a_2 \\ \vdots \\ a_N \end{bmatrix} = \begin{bmatrix} a_1 \\ \vdots \\ a_{N-1} \end{bmatrix} c + \begin{bmatrix} \epsilon_2 \\ \vdots \\ \epsilon_N \end{bmatrix} \tag{11}$$

Estimate c is calculated according to the method of least squares [60]; the calculation of the estimate of correlation coefficient R is well known

$$c = \frac{\sum_{i=1}^{N-1} a_i^2}{\sum_{i=1}^{N-1} a_i a_{i+1}}, \qquad R = \frac{\sum_{i=1}^{N} (a_i - m_i)(a_{i+1} - m_{i+1})}{\left[\sum_{i=1}^{N} (a_i - m_i)^2\right]^{1/2} \left[\sum_{i=1}^{N} (a_{i+1} - m_{i+1})^2\right]^{1/2}} \tag{12}$$

From the shape of these two parameter calculations of the estimates, comes their similar course. Again, we selected two extreme records, LR3 and N6, Figure 14:

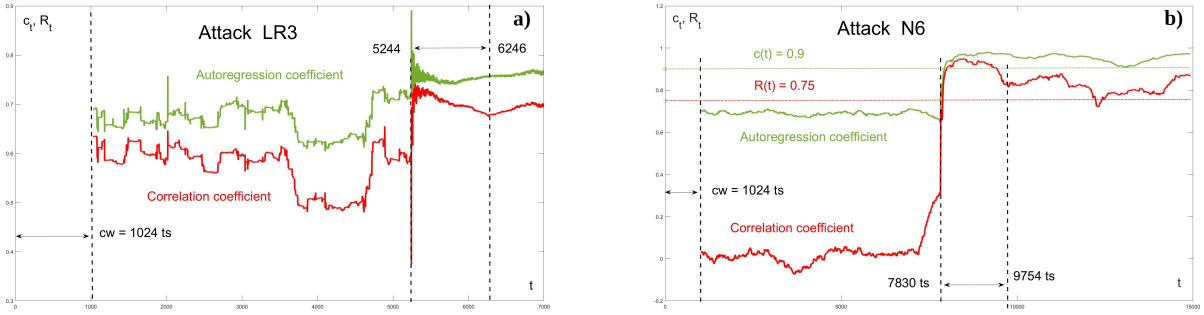

Figure 14. Autoregression and correlation coefficients. (**a**) Attack LR3. (**b**) Attack N6.

At the moment of the starting point of the attack, the values of the coefficients began to grow rapidly toward the value of 1 for most of the records. Thanks to this, we can set a certain limit value (threshold) for both coefficients, the crossing of which would signal a DDoS attack. The lower the value, the more timely the attack signaled; on the other hand, very low attacks can cause false reports. Based on the performed experiments, we set the limit for the moving autoregressive coefficient to $c(t) = 0.9$, and the moving correlation coefficient to $\varrho(t) = 0.75$. For most recordings, except for LR3 and P4, Figures S24 and S28, thresholds set in this way can be used for machine detection.

4.2.2. Kullback–Leibler Divergence

Another quantity that uses two computing windows is the Kullback–Leibler divergence KD. Divergence is one of the measurements used in mathematical statistics to determine how one probability distribution function (P) differs from another probability distribution function (Q).

The attack is it compares the probability distribution of two stochastic variables $X_t(\omega)$ and $X_{t-1}(\omega)$. The vectors of the realization of these two variables in successive overlapped compute windows of size N are known as $\mathbf{a}_t$ and $\mathbf{a}_{t-1}$. We denote the divergence estimate KD as dvg. Let n_k be the number of increments a_i in the compute window $\mathbf{a}_{t-1}$ and m_k in the next $\mathbf{a}_t$. For the calculation of the divergence KD and its estimate dvg, we have relations

$$KD(P\|Q) = \sum_{k=0}^{m} P(k) \ln \frac{P(k)}{Q(k)}, \qquad dvg = \sum_{k=0}^{m} \left[\frac{n_k}{N}\right] \ln \left[\frac{n_k}{m_k}\right] \tag{13}$$

Divergence reacted to most records immediately when loading the first compute window with offensive traffic with a high impulse. However, it had a tendency to react in this way to peaks in normal traffic, which could cause a lot of false reports during detection. For example, in attack S1, the impulse during the attack is indistinguishable from impulses during normal traffic, Figure 15a. In attack LR3, the course of divergence is completely ideal for machine recognition, Figure 15b:

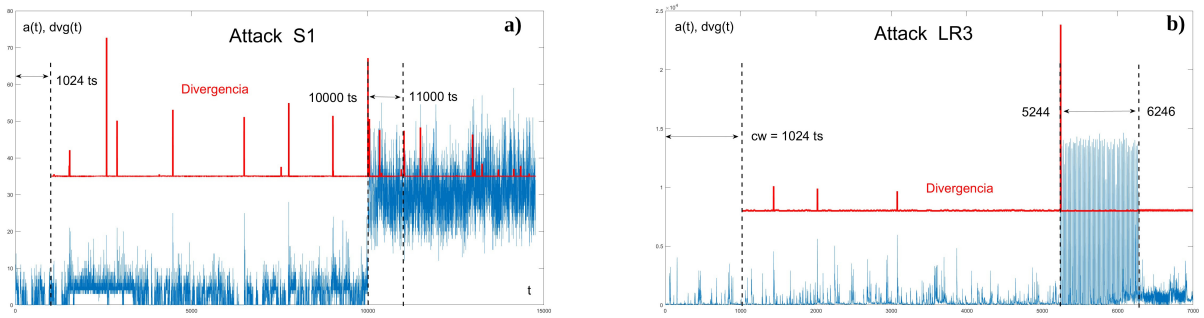

Figure 15. Kullback–Leibler divergence. (**a**) Attack S1. (**b**) Attack LR3.

Another disadvantage of using divergence is the fact that the high impulsion at the start of the attack lasts a very short time compared to other parameters, so we see its application in combination with other parameters as problematic. We have, therefore, postponed the use of divergence for the next research.

4.3. Effectiveness of the Use of Statistical Coefficients

In the following Table 1, we have summarized the evaluation of the studied statistical coefficients' reaction to the analyzed DDoS attack captures. From the three studied estimates of the Hurst exponent, we selected the estimate using RS analysis with the removal of the autoregressive linear trend. We considered this estimate to be the most effective.

Table 1. Effectiveness of statistical coefficients, coefficient of variation $V(t)$, kurtosis $K(t)$, skewness $Skw(t)$, Hurst exponent $H(t)$, autoregressive coefficient $c(t)$, correlation coefficient $R(t)$, Kullback–Leibler divergence $dvg(t)$.

Attack	$V(t)$ 1.00	$K(t)$	$Skw(t)$	$H(t)$ 1.00	$c(t)$ 0.90	$R(t)$ 0.75	$dvg(t)$
N2	0	PT	PT	9	PT	PT	4
N6	0	PT	PT	0	PT	PT	5
N7	0	x	x	1	PT	PT	4
N8	0	PT	PT	0	PT	PT	0
S1	5	x	x	6			8
S5	0	PT	PT	0	PT	PT	0
LR3	PT	PT	PT	x	PT	PT	3
P4	x	x	x	PT	x	x	x

Green cells mean that the attack was recognized by exceeding the thresholds of the given parameters. Threshold values are listed in the right row of the table. The number in the cell means the number of false reports and reps. It exceeds the limit value before the attack. Especially in the case of divergence, this means the number of counter impulses before the actual attack.

- The designation "PT" means that with the given parameters, we assume that the attack could be recognized using prediction methods (PT, predicting tunnel [8]);
- The marking "x" means that the given parameter is not applicable for the given record.

We can divide the parameters into two groups based on the performed experiments. In the first group, we included the parameters for which we can use their threshold value for attack detection: $V(t)$, $H(t)$, $c(t)$, and $R(t)$. The second group includes parameters for which prediction methods must be used to detect an attack, mainly $K(t)$, $Skw(t)$, and also $c(t)$ and $R(t)$. We see that the division into groups is not clear-cut. Divergence, due to the short impulse duration during the attack and frequent reactions to peak peaks in normal traffic, we have excluded from further considerations.

5. Predicting σ-Tunnel

The idea of a simple prediction σ-Tunnel, determined using average and deviation values of the given parameter, was presented in [8]. Next, we dealt with tunnel creation using a polynomial regression model, Fourier transformation, and autoregression analysis. However, the effectiveness of these compared to computationally demanding methods, σ-Tunnel, was significantly worse, not only with a later time of reporting the attack, but also with the occurrence of several times more false positives [61]. Therefore, we will only deal with the σ-Tunnel.

For the θ parameter, we determine the prediction window of size $|pw| = N$. We mark the parameter values in the window with $\theta_1, \ldots, \theta_N$. From the θ_t values, we calculate the average $\overline{\theta}$ and standard deviation σ_θ. We will create an interval around average $\overline{\theta}$, $I = \langle \overline{\theta} - n\sigma, \overline{\theta} + n\sigma \rangle$ and then test whether the new value of the parameter belongs to the predicting interval, $\theta_{N+1} \in I$. If it does not, the machine detects the beginning of the

attack. Next, we shift the prediction window pw by one parameter value θ and repeat the whole process.

$$\overline{\theta} = \frac{1}{N}\sum_{i=1}^{N}\theta_i, \qquad \sigma^2 = \frac{1}{N-1}\sum_{i=1}^{N}\left(\theta_i - \overline{\theta}\right)^2 \qquad \text{Test:} \quad \theta_{N+1} \in \left\langle \overline{\theta} - n\sigma, \overline{\theta} + n\sigma \right\rangle \tag{14}$$

Based on the experiments, we decided to use a prediction window of size $|pw| = 1000$ and the width of the interval $I = \left\langle \overline{\theta} - 3\sigma, \overline{\theta} + 3\sigma \right\rangle$. With such settings, the prediction tunnel was able to adapt to the development of parameter values and detect sudden changes at the start of the DDoS attack, with only a relatively small number of false reports.

When using the prediction tunnel directly on increments of the IP flow a_t, we found that although exceeding the upper limit of the tunnel detects the starting point of a DDoS attack relatively early, at the same time, there are frequent false reports. When the test interval increased, the number of false reports decreased, but at the same time, the ability of the method to detect an attack decreased. We selected records N8 and N1 as an example. Only the upper limit of the 3σ-Tunnel is shown in Figure 16.

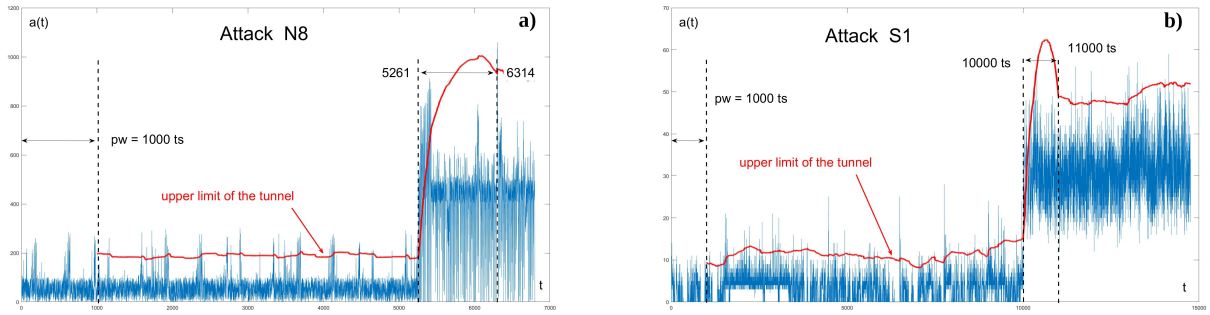

Figure 16. The upper limit of predicting the tunnel for incremets a_t of records: (**a**) N8, (**b**) S1.

Using statistical parameters to detect DDoS attacks enables the suppression of peaks in the IP flow thanks to a relatively large compute window.

In the following figures, we will demonstrate the behavior of the 3σ-Tunnel on the parameter with the greatest variability among the considered statistical coefficients, namely on the Hurst exponent (estimated by RS statistics with removing the autoregressive trend). In Figure 17a, there is an ideal case where the detection occurred when loading the third time slot with an offensive traffic, and no false reports occurred. In Figure 17b, the attack was not detected, and there was one false report:

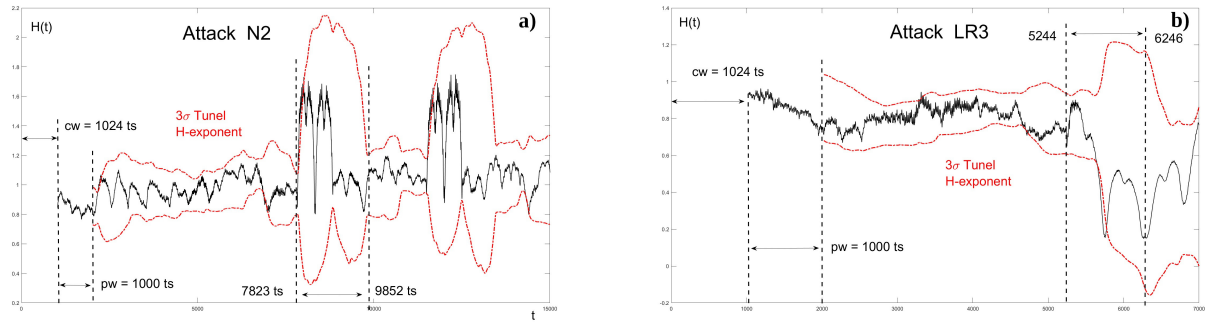

Figure 17. The 3σ-Tunnel for Hurst exponent (without RS). (**a**) N2, (**b**) LR3.

Calculating the Hurst exponent is significantly more time-consuming than other statistical parameters, so we have temporarily postponed its use for detection.

Based on the experiments performed with different tunnel widths and different prediction window sizes, we can say that the 3σ-Tunnel with $|pw| = 1000\ ts$ has a sufficiently "long memory" to be able to cover the local variability (jitter) of the statistical coefficient

without causing a large number of false reports. At the same time, the prediction tunnel set up in this way has a sufficiently "long memory" to be unable to react to a significant increase in the values of the given coefficient at the start of a DDoS attack.

In Table 2, we present detections using the 3σ-Tunnel applied to selected statistical coefficients. The F/R symbol represents the number of false reports (F) and the attack detection time (R) in *ts*. For example, the first value in Table 2, 2/30, means that when using the 3σ-Tunnel applied to the variation $V(t)$, there was an attack recognized in the record S1 within 30 *ts* of its beginning. Before that, there were two false reports (we have excluded the problematic record P4 from the experiments for now and will devote a separate subsection to it).

Table 2. Detection using 3σ-Tunnel applied to: variability $V(t)$, skewness $K(t)$, kurtosis $Skw(t)$, autoregression $c(t)$, and correlation $\varrho(t)$.

Attack	$V(t)$	$K(t)$	$Skw(t)$	$c(t)$	$\varrho(t)$
S1	2/30	1/5	2/6	1/9	1/9
N2	1/116	5/24	2/47	0/1	1/1
LR3	0/1	2/1	1/1	1/1	1/8
S5	4/18	4/20	3/42	7/1394	6/5
N6	3/1	2/4	1/10	3/17	5/13
N7	1/51	2/40	2/40	1/90	2/76
N8	0/2	1/1	0/1	5/2	0/3

We consider the R parameter (the attack detection time) to be more important than the parameter F (the number of false reports) because a suitable combination of statistical coefficients can eliminate the number of false reports. Therefore, we will use both parameters to compare the effectiveness of the coefficients. We will use the PCA (Principal Component Analysis) method to visualize the similarity in the 3D view [62] as shown in Figure 18. In the data stage of the method, two matrices $\mathbf{B}_F$ and $\mathbf{B}_R$ of dimensions 6×7 are represented, which contain the values of the parameters F and R from Table 2. Unlike the table, the rows of the matrix represent statistical coefficients from $V(t)$ to $\varrho(t)$ and the column records from S1 to N8 (transposed table):

$$\mathbf{B}_F = \begin{pmatrix} 2 & 1 & 0 & 4 & 3 & 1 & 0 \\ 1 & 5 & 2 & 4 & 2 & 2 & 1 \\ 2 & 2 & 1 & 3 & 1 & 2 & 0 \\ 1 & 0 & 1 & 7 & 3 & 1 & 5 \\ 1 & 1 & 1 & 6 & 6 & 2 & 0 \end{pmatrix} \quad \mathbf{B}_R = \begin{pmatrix} 30 & 116 & 1 & 18 & 1 & 51 & 2 \\ 5 & 24 & 1 & 20 & 4 & 40 & 1 \\ 6 & 47 & 1 & 42 & 10 & 40 & 1 \\ 9 & 1 & 1 & 1394 & 17 & 90 & 2 \\ 9 & 1 & 18 & 5 & 13 & 76 & 3 \end{pmatrix} \tag{15}$$

For 3D visualization, we transform line vectors matrices $\mathbf{B}_F$ and $\mathbf{B}_R$ into spaces with Karhunen–Loev base $\mathbf{U}_F$ and $\mathbf{U}_R$ (orthonormal basis of eigenvectors):

$$\mathbf{C}_F = \mathbf{B}_F\mathbf{U}_F, \qquad \mathbf{C}_R = \mathbf{B}_R\mathbf{U}_R \tag{16}$$

For the 3D visualization, we use the first three columns of the matrix $\mathbf{C}_F$ and $\mathbf{C}_R$ (the first three main components), Figure 18:

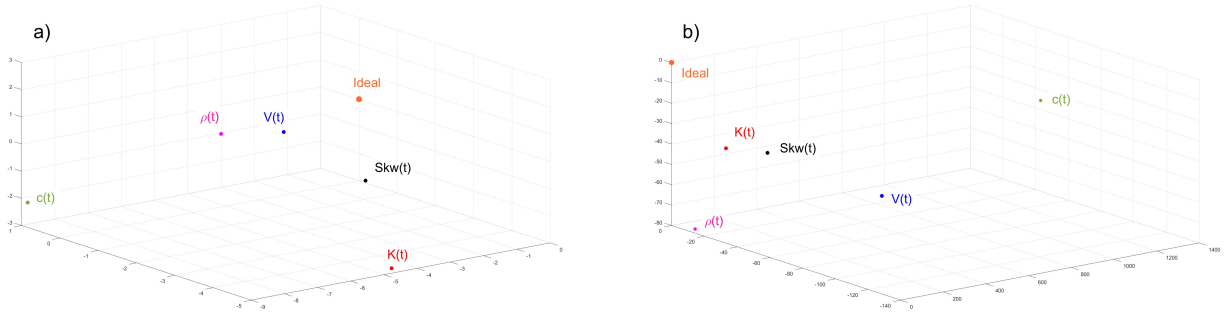

Figure 18. The 3D visualization of parameter values from Table 2. (**a**) F (False), (**b**) R (Recognize) Attack.

The effectiveness of individual statistical coefficients is determined according to the distance from the zero vector, Table 3. For the F parameter, the zero vector represents the zero number of false reports for each traffic capture, and the R parameter represents the recognition of the attack before it started.

Table 3. Effectiveness of the use of coefficients in false reports and attack recognition.

Parameters	$V(t)$	$K(t)$	$Skw(t)$	$c(t)$	$\varrho(t)$
F (False)	0.0173	0.0026	0.0056	1.9517	0.0064
R (Recogn.)	28.05	54.92	19.07	85.35	78.92

In the case of false reports, the order of coefficients from the most effective is $K(t)$, $Skw(t)$, and $\varrho(t)$. The order of attack detection speed is $Skw(t)$, $V(t)$, and $K(t)$. The worst was the autocorrelation coefficient $c(t)$. We will use these results in the next chapter to create detection functions.

6. Detection Functions for Machine Recognition of DDoS Attacks

In the previous chapters, we outlined two possible ways of recognizing DDoS attacks using statistical coefficients. The first way is the use of threshold values for appropriate coefficients. The second way is applying the prediction tunnel to the course of the coefficient values.

6.1. Detection Method Using Threshold Values

For the Detection method using threshold values (DTV), the following statistical parameters are useful: coefficient of variable $V(t)$, Hurst exponent $H(t)$, autoregressive and correlative coefficients $c(t)$ and $\varrho(t)$. For $V(t)$ and $H(t)$, it is a value of 1.00; based on performed experiments, we proposed a value of 0.90 for $c(t)$ and for $\varrho(t)$ the value 0.75. Our effort is to create a computationally simple detection method. That is why we left out Hurst's exponent from further consideration.

We introduce a two-valued 0/1 logic function $\mathrm{Y}(t)$, which evaluates the exceedance threshold tr for a given statistical coefficient θ:

$$\mathrm{Y}_\theta(t) = \mathrm{Y}(\theta(t) \geq tr) = 1, \qquad \mathrm{Y}_\theta(t) = \mathrm{Y}(\theta(t) < tr) = 0 \tag{17}$$

In the following graphs, we will show offensive traffic captures S1 and N2 for a course of coefficients $V(t)$, autoregressive coefficient $c(t)$, and correlation coefficient $\varrho(t)$, Figure 19, and the course of their functions $\mathrm{Y}_V(t)$, $\mathrm{Y}_c(t)$, and $\mathrm{Y}_\varrho(t)$, Figure 20 :

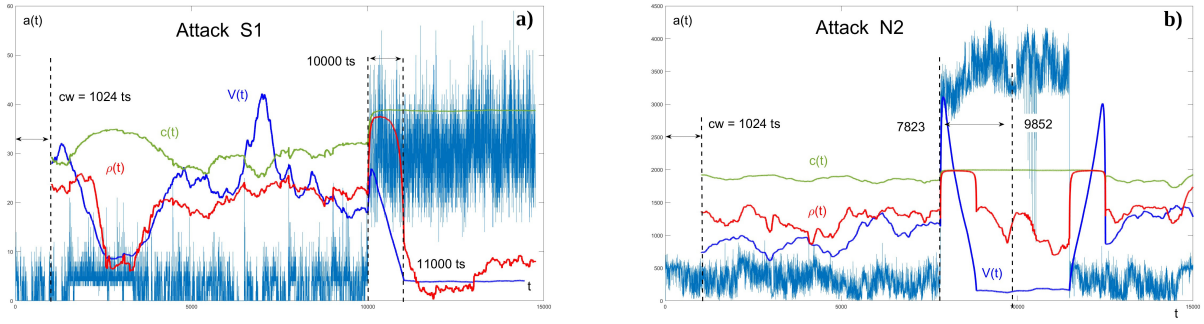

Figure 19. Variation, autoregressive, and correlation coefficients. (**a**) Record S1, (**b**) Record N2.

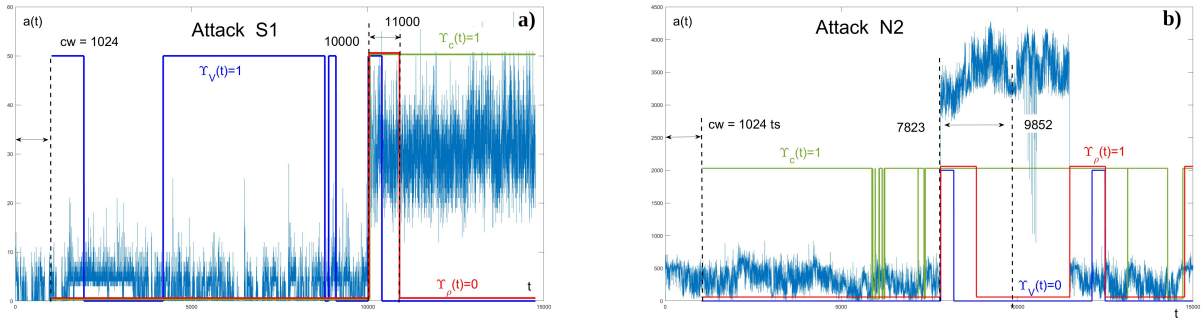

Figure 20. Logical functions $Y_V(t)$, $Y_c(t)$, and $Y_\varrho(t)$. (**a**) Record S1, (**b**) Record N2.

Sets of values on which coefficients exceed their threshold are referred to as detection intervals. In Figure 20, we see how these intervals overlap each other. In order to eliminate false reports and, at the same time, achieve timely detection of attacks, we will introduce a three-valued detection function $D_V(t)$:

$$D_V(t) = Y_V(t) \cdot Y_c(t) + Y_V(t) \cdot Y_\varrho(t) + Y_c(t) \cdot Y_\varrho(t) \tag{18}$$

The course of the detection function for records S1 and N2 is in Figure 21:

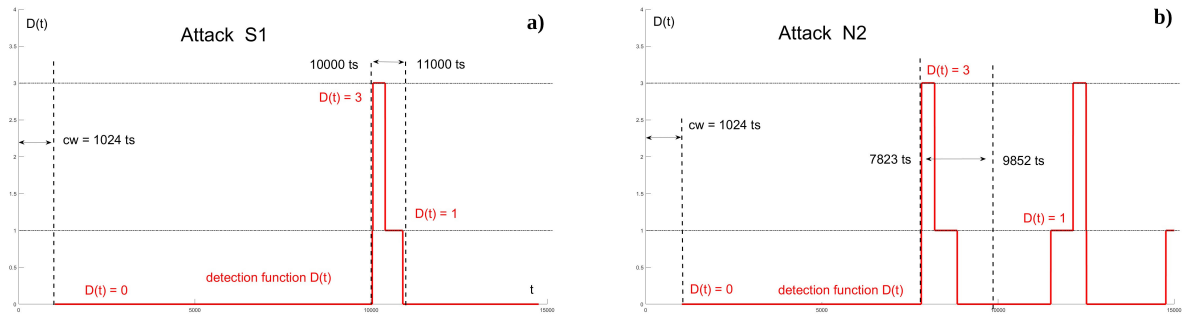

Figure 21. Detection function $D_V(t)$. (**a**) Record S1, (**b**) Record N2.

In both traffic captures, the coefficient values exceeded their thresholds already during standard traffic processing in the case of record S1 variation coefficient, Figure 20a, and in the case of record N2 autoregressive coefficient, Figure 20b. However, thanks to the shape of the detection function (20), false reports were eliminated. According to the values $D_V(t) = 1$ and $D_V(t) = 3$, we can determine in which time slot two or all three statistical coefficients detected the attack.

In Table 4, we present the evaluation of the use of detection $D_V(.)$ functions for the analyzed attacks. The first value represents the number of false reports, the second determines the time in which the attack was detected after the start of the attack of at least

two coefficients, and the third value indicates the time when all three coefficients $V(t)$, $c(t)$, and $\varrho(t)$. The sign "x" means that the detection function did not recognize the attack for the given recording (statistical coefficients did not exceed their thresholds at the beginning of the attack).

Table 4. Evaluation of attack recognition using the detection function $D(t)$.

DTV	S1	N2	LR3	S5	N6	N7	N8
$D_V(t)$	0/34/34	0/5/5	x	x	0/75/119	0/228/480	0/47/210

The detection function $D_V(.)$ was able to eliminate false reports for all the examined records. It successfully detected the attack, especially with standard attacks; with LR3, P4, and S5 records, in general, it did not react to the attack. Another shortcoming of the method may be relatively late attack detection, e.g., in recording, the N7 attack started at 8300 *ts*, the detection function recognized it with a value of 1 at 8528 *ts* (228 ts from attack) and with a value of 3 at 8780 *ts* (480 *ts* from attack), Table 4. On the other hand, the local maximum for the moving average rate was recorded at 9680 *ts* (1380 *ts* from attack). Our effort is to find additional detection functions that would achieve better results regarding attack detection time.

6.2. Detection Method Using Predicting Tunnel

In Section 5, we determined the first three most effective statistical coefficients concerning the speed of attack propagation using 3σ-tunnels: kurtosis $K(t)$, skewness $Skw(t)$, and variations $V(t)$. Since we have selected coefficients whose values increase significantly when a DDoS attack starts, we will only deal with the upper limit of 3σ-tunnels. The time during which the given coefficient θ exceeds the upper limits of tunnel $ul(t)$ is denoted as the detection interval and is described by the function $G(t)$:

$$G_\theta(t) = G(\theta(t) \geq ul(t)) = 1, \qquad G_\theta(t) = G(\theta(t) < ul(t)) = 0 \tag{19}$$

In Figure 22, we show the history of the function $G(.)$ on records S1 and N7. Together with increments of flow $a(t)$, we will show the course of kurtosis coefficient $K(t)$, upper limits of 3σ-Tunnel of $K(t)$, and the corresponding function $G_K(t)$:

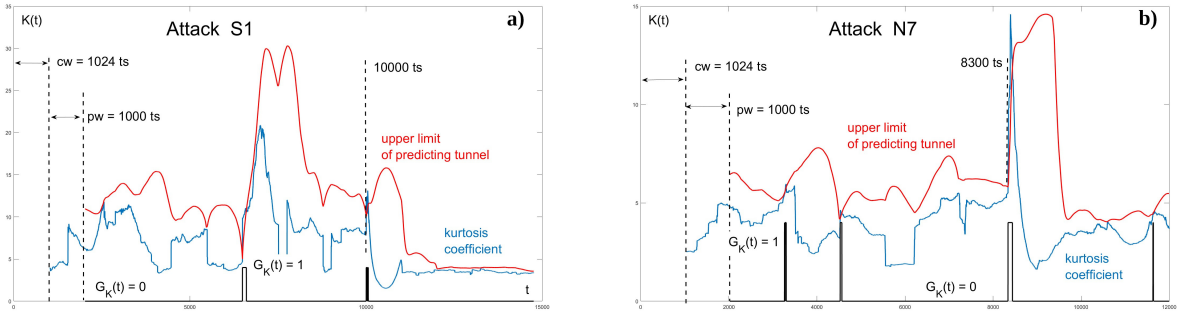

Figure 22. Incremets of flow $a(t)$, kurtosis coefficient $K(t)$, upper limits of 3σ-Tunnel of $K(t)$, and function $G_K(t)$ for (**a**) Record S1, (**b**) Record N7.

In both records, the kurtosis coefficient exceeded the upper limit of 3σ-Tunnels already during normal IP traffic processing. To eliminate false reports, we use the same form of the detection function as in the previous method, which used thresholds of statistical coefficients. We denote the detection function for the detection method using the predicting tunnel (DPT) by $D_T(t)$:

$$D_T(t) = G_K(t) \cdot G_{Skw}(t) + G_K(t) \cdot G_V(t) + G_{Skw}(t) \cdot G_V(t) \tag{20}$$

In Table 5, we present the evaluation of the use of detection functions $D_T(.)$. The method of evaluation is identical to the evaluation of the function $D_V(t)$ in Table 4. The first value represents the number of false reports, the second determines the time of attack detection by at least two coefficients, and the third value represents detection by all three coefficients.

Table 5. Evaluation of attack recognition using the detection function $D_T(t)$.

DPT-3σ	S1	N2	LR3	S5	N6	N7	N8
$D_T(t)$	1/6/28	0/1/1	1/1/1	1/18/41	1/12/12	1/39/49	0/1/1

Compared to the previous method, the $D_T(t)$ detection function was able to recognize the starting point of a DDoS attack for all analyzed traffic captures, even significantly earlier than the $D_V(t)$ function. However, the method also has a disadvantage: with almost all records, there was one false report. Therefore, the next direction of our research was the effort to eliminate false reports while maintaining the early detection of an attack.

6.3. Detection Method Using Holt-Exponential Smoothing

In an effort to eliminate false reports, we decided to apply smoothing methods to the course of statistical coefficient values. The simplest method is exponential smoothing [63].

It is a relatively simple method in which the values of the statistical coefficient $\theta(t)$ are replaced by their weighted average $s(t)$ according to the relationship:

$$s(1) = \theta(1), \qquad s(t) = \alpha \cdot \theta(t) + (1-\alpha) \cdot s(t-1), \qquad 0 < \alpha < 1, \quad t = 2,3,\ldots \tag{21}$$

The value $s(t)$ represents the so-called exponentially weighted moving average in time t. The value of α determines the degree of smoothing if $\alpha \to 1$, the smoothing is minimal and $s(t) \doteq \theta(t)$. In case $\alpha \to 0$ results in strong smoothing, and the method minimally reacts to local fluctuations in the values of the coefficient $\theta(t)$. We can edit the relationship (21):

$$s(t) = \alpha \cdot \theta(t) + \alpha(1-\alpha) \cdot \theta(t-1) + \alpha(1-\alpha)^2 \cdot \theta(t-2) + \ldots + (1-\alpha)^{t-1} \cdot \theta(1) =$$

$$= \alpha \sum_{k=0}^{t-2} (1-\alpha)^k \cdot \theta(t-k) + (1-\alpha)^{t-1} \cdot \theta(1), \quad t = 2,3,\ldots \tag{22}$$

We already presented the first experiments with exponential smoothing in [8]. When performing experiments on other records, the shortcomings of this smoothing became apparent. Significant suppression of local peaks in courses of the examined coefficients was manifested at the value of the weight parameter around $\alpha = 0.05$. A time series smoothed in this way has a "long-term" memory and only minimally adapts to the newly read values. Although there were local peaks smoothed out, at the same time, the values of the coefficients have shifted significantly over time, and in some records, even the growth of values of the coefficients was suppressed at the moment of the start of the DDoS attack, for example, Figures 23 and 24.

A more complicated smoothing method is Holt-exponential smoothing, $h(t)$, and double exponential smoothing [64]. Another smoothing parameter $0 < \beta < 1$ and the component $k.b(t), R \in$ are added to the model, which represents the estimate of the linear trend of $\theta(t)$ values at time t. Since, in our case, we do not know in advance what trend the smoothed time series $\theta(t)$ will have, we put $b(1) = 0$:

$$a(1) = \theta(1), \qquad b(1) = 0, \qquad h(1) = a(1) + kb(1) = \theta(1) \tag{23}$$

$$a(t) = \alpha \cdot \theta(t) + (1-\alpha) \cdot h(t-1), \qquad b(t) = \beta \cdot [\theta(t) - \theta(t-1)] + (1-\beta) \cdot b(t-1) \tag{24}$$

$$h(t) = s_H(t) + kb(t) \qquad t = 2,3,\ldots \tag{25}$$

After the adjustments, we can write the individual components of Holt's smoothing as follows:

$$a(t) = \alpha \sum_{k=0}^{t-2} (1-\alpha)^k \cdot \theta(t-k) + (1-\alpha)^{t-1} \cdot \theta(1) + \sum_{k=1}^{t-2} k(1-\alpha)^k \cdot b(t-k) \tag{26}$$

$$b(t) = \beta \sum_{k=0}^{t-2} (1-\beta)^k \cdot [\theta(t-k) - \theta(t-k-1)], \qquad t = 2, 3, \ldots \tag{27}$$

Using the traffic captures of the DDoS attack, we performed comparisons of the influence of exponential smoothing and Holt-exponential smoothing of statistical coefficients on the effectiveness of detection using the predicting tunnel DPT. We compared Holt-exponential smoothing with different settings of the parameters α, β, and k with exponential smoothing with the same smoothing parameter α. Finally, after subjective evaluation, we chose the values $\alpha = 0.05$, $\beta = 0.8$, and $k = 0.8$. With such a setting, Holt-exponential smoothing was able to smooth out the local peaks of the given coefficient and, at the same time, did not significantly deviate from the overall course of values, as was achieved in the case of exponential smoothing with the same smoothing coefficient $\alpha = 0.05$.

Figure 23 displays the example of both smoothing methods used for the kurtosis coefficient $K(t)$ and correlation coefficient $\varrho(t)$:

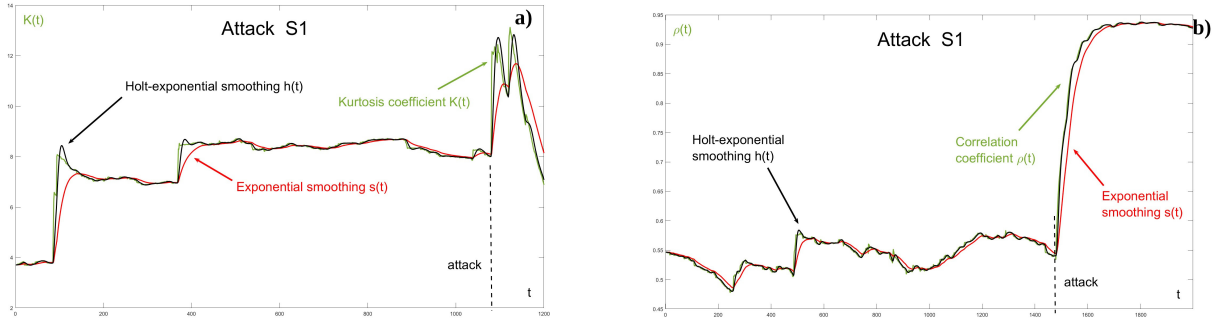

Figure 23. Record S1. Exponential smoothing with $\alpha = 0.05$ and Holt-exponential smoothing with $\alpha = 0.05$, $\beta = 0.8$, and $k = 0.8$. (**a**) Kurtosis coefficient $K(t)$. (**b**) Correlation coefficient $\varrho(t)$.

Even though both smoothing methods have the same "memory" length ($1 - \alpha = 0.95$), thanks to which significant local peak suppression occurs, exponential smoothing causes a significant time delay. Thanks to the inclusion of a trend component in the Holt-exponential smoothing, such an effect does not occur.

The next Figure 24 displays autoregressive coefficient smoothing $c(t)$ by both methods for records S1 and LR3:

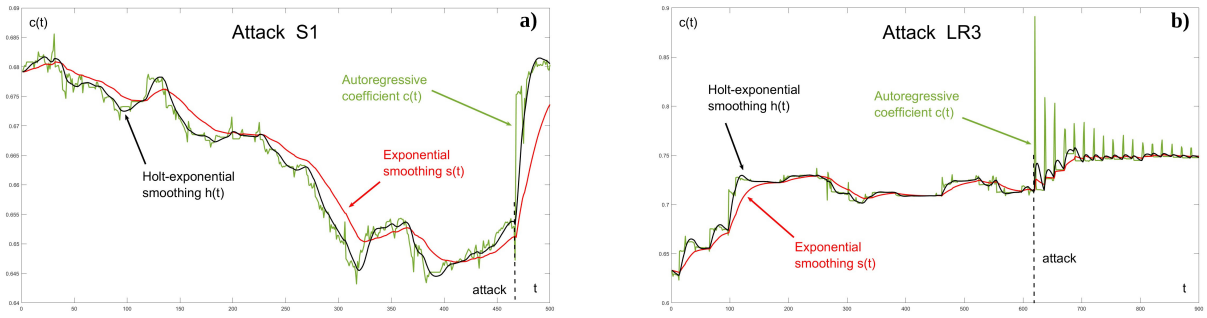

Figure 24. Exponential smoothing $\alpha = 0.05$ and Holt-exponential smoothing $\alpha = 0.05$, $\beta = 0.8$, and $k = 0.8$ of autoregressive coefficient $c(t)$: (**a**) record S1, (**b**) record LR3.

In Figure 24a, we see that exponential smoothing of the autoregressive coefficient with record S1 goes completely outside the values range of the coefficient. Using traffic capture LR3 (Figure 24b), even exponential smoothing completely suppressed the peaks in the course of the autoregressive coefficient at the start of a DDoS attack. The Holt method also significantly reduced peaks, but still remained recognizable from the previous course of the coefficient.

Before using the DPT prediction tunnel detection method, we first smoothed the considered coefficients of variation, kurtosis, and skewness using the Holt-exponential method and only after the smoothing we used the predictive 3σ-tunnel. We called this method the Detection Method using Predictive Tunnels with Holt-exponential smoothing (DPT-Hs). Two-valued decision function signaling in the crossing of the upper limit prediction interval for individual smoothed coefficients is denoted as $H(.)$, and the detection function of the DTV-Hs method itself is $D_H(t)$:

$$D_H(t) = H_K(t) \cdot H_{Skw}(t) + H_K(t) \cdot H_V(t) + H_{Skw}(t) \cdot H_V(t) \tag{28}$$

In Table 6, we present the evaluation of the detection functions $D_H(.)$ usage. The method of evaluation is identical to the evaluation of the function $D_T(t)$ shown in Table 5. The first value represents the number of false reports, the second determines the time of attack detection by at least two coefficients, and the third value determines detection by all three coefficients.

Table 6. Evaluation of attack recognition using the detection function $D_H(t)$.

DPT-Hs-3σ	S1	N2	LR3	S5	N6	N7	N8
$D_H(t)$	1/39/61	0/3/3	0/9/12	0/39/39	0/13/13	0/68/81	0/13/21

Except for the non-standard S1 attack, we noticed on all other traffic captures that the use of Holt-exponential smoothing on detection using the predictive tunnel eliminated all of the false reports. The smoothing, however, caused a time shift in the values of the used statistical coefficients, which resulted in a slight delay in signaling the start of a DDoS attack. The elimination of false reports at the expense of a slight delay in DDoS attack signaling leaves an open question for the process of real deployment in IP traffic monitoring.

7. Detection of Problematic P4 Attack

In previous chapters, we identified the P4 attack as problematic. The reason was that based on the course of the values of the considered coefficients, this attack was unrecognizable using the detection methods presented so far. Of all the analyzed DDoS attacks that traffic captures, we are more interested in the examination of this attack since it was an attack on the network infrastructure of our own University. Therefore, we continued to search for other detection methods to recognize this problematic attack.

We can consider this attack, according to the course of the moving average rate, as a low-rate DDoS attack (when attacks do not occur periodically), Figure 7b. Marking the beginning of the attack by the admin is in 3060 *ts* (1 *ts* = 10 *ms*), but from the record of increments a certain change in the nature of the flow can be recognized as early as 2650 *ts*. At this moment, the attack itself was not recognized by the detection function $D(t)$ using threshold values of coefficients, or by the functions $D_T(t)$ and $D_H(t)$ with the predictive 3σ-tunnel. After conducting experiments with different statistical coefficients and different multiples of the $n.\sigma$-tunnel, we found that with 2σ-tunnel, some coefficients recognized the beginning of the attack after 3060 ts, Figure 25, and some coefficients reacted to the change in the nature of the IP flow in the interval $\langle 2650, 3060 \rangle$, Figure 26:

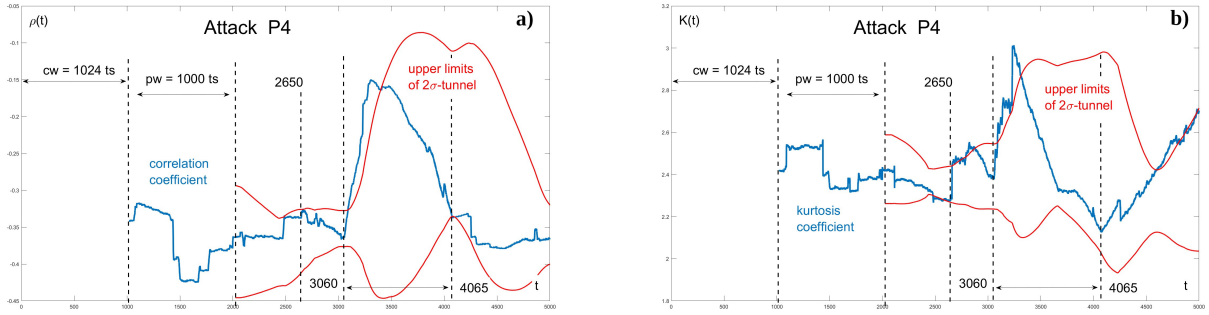

Figure 25. Record P4, 2σ-tunnel for: (**a**) correlation coefficient $\varrho(t)$, (**b**) kurtosis coeffiecient $K(t)$.

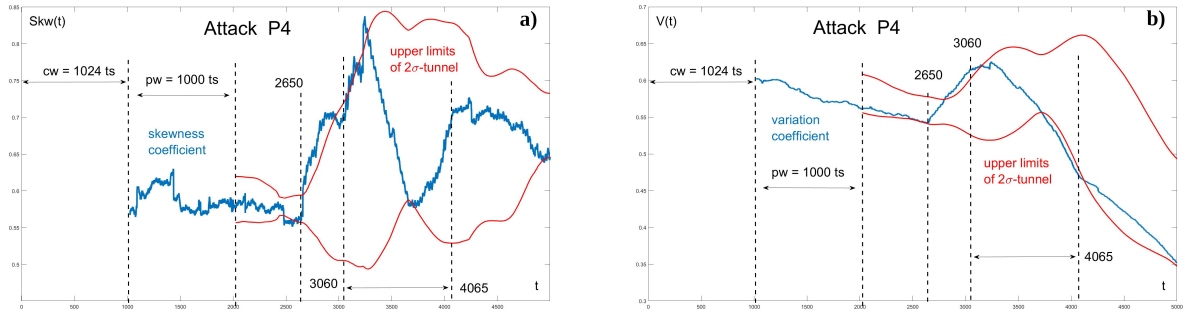

Figure 26. Record P4, 2σ-tunnel for (**a**) skewness coefficient $Skw(t)$, (**b**) variation coefficient $V(t)$.

DPT-2σ method with detection function $D_T(t)$ using coefficients $V(t)$, $K(t)$, and $Skw(t)$ with other records caused a lot of false positives. In an effort to find an acceptable solution for all the tested traffic captures, we conducted further experiments with all record combinations of various statistical coefficients and parameters of Holt-exponential smoothing by different forms of the detection function $D_H(t)$. Based on our subjective evaluation, we have devised a method that relatively satisfactorily recognizes all of the investigated attacks. The method consists of the following steps:

- Calculation of moving coefficients of variation $V(t)$, kurtosis $K(t)$, skewness $Skw(t)$, and correlation $\varrho(t)$;
- Values of all coefficients are smoothed using Holt-exponential smoothing;
- Signaling of exceeding the upper limit 2σ-tunnel using the function $H_\theta(t)$;
- DDoS-attack detection using the multiplicative detection function $D_M(t)$:

$$D_M(t) = H_V(t) \cdot H_K(t) \cdot H_{Skw}(t) \cdot H_\varrho(t) \tag{29}$$

The course of coefficients $V(t)$, $K(t)$, $Skw(t)$, $\varrho(t)$, and multiplicative detection function is shown in Figure 27:

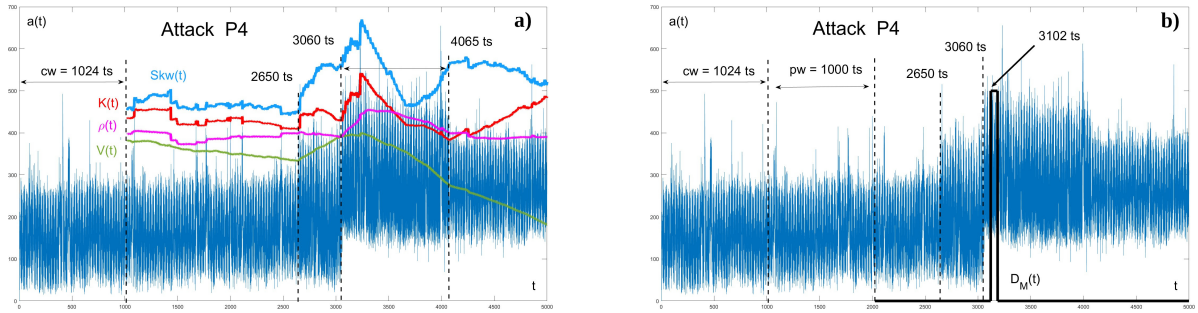

Figure 27. (**a**) The coefficients $V(t)$, $K(t)$, $Skw(t)$ and $\varrho(t)$ for record P4, (**b**) multiplicative detection function $D_M(t)$ for record P4.

Holt-exponential smoothing reduced the course value variance of the considered coefficients. By changing the size of the prediction interval to 2σ, using these coefficients, we achieved an attack signaling and multiplicative form $D_M(t)$, as a product of decision functions $H_\theta(t)$, which eliminated false reports.

In Table 6, we present the evaluation of the usage of detection functions $D_M(.)$. Unlike the previous tables, in Table 7, since the $D_M(t)$ function is only 0/1-valued, we include only two values, the first represents the number of false attacks, as before, and the second considers the moments of detection of attacks after its beginning in [ts]:

Table 7. Evaluation of attack recognition using the detection function $D_H(t)$.

DPT-Hs-2σ	S1	N2	LR3	S5	N6	N7	N8
$D_M(t)$	1/51	0/7	0/43	1/41	0/−106	0/92	0/26

The method detected an attack at every examined record, at S1 and S5 it signaled a false report, and sample N6 recognized the onset of the attack 106 ts earlier. It is questionable whether to evaluate this event as a false positive or as an early detection of an attack.

8. Results and Discussion

Our effort is to create an autonomous self-learning system of relatively simple software-implementable statistical methods for the purpose of early detection of DDoS attacks usable in high-speed network infrastructure. The system must record a change from standard traffic to attack traffic in time.

We tested several different statistical coefficients that describe the probabilistic structure of the network flow. The reaction of tested coefficients on offensive traffic was tested on eight selected records of DDoS attacks of various types.

Based on the response of statistical coefficients to the onset of a DDoS attack, we divided the coefficients into two groups.

In the first group, we included the parameters with which we can detect the attack using their threshold value. We have the coefficient of variation $V(t)$ (1.00), Hurst exponent $H(t)$ (1.00), autoregressive coefficient $c(t)$ (0.90), and correlation coefficient $\varrho(t)$ (0.75).

The second group includes parameters for which prediction methods must be used to detect an attack. Here, we included the kurtosis and skewness coefficients $K(t)$, $S(t)$, and autoregressive and correlation coefficients $c(t)$, $\varrho(t)$. Later we also included the coefficient of variation $V(t)$. We see that the division into groups is not clear-cut. The divergence $dvg(t)$ was due to the short duration of the impulse during the attack and also the frequent reactions to peaks in standard traffic excluded from further considerations.

From various prediction methods, which we do not mention in this article, we chose the so-called predicting $n\sigma$-Tunnel, which uses the average value of the coefficient and its standard deviation σ based on experiments with different tunnel widths and different sizes

Using the PCA method [65], we evaluated statistical coefficients according to the number of false reports and the speed of attack recognition. In the case of false reports, the order of the coefficients is from the most effective $K(t)$, $Skw(t)$, and $\varrho(t)$. The order of attack detection speed is $Skw(t)$, $V(t)$, and $K(t)$. The worst in both cases turned out to be the autocorrelation coefficient $c(t)$. We used the results when creating detection functions.

We have divided the detection methods intended for machine recognition of a DDoS attack into the Detection Method using threshold values (DTV) and the Detection Method using the Predicting Tunnel (DPT). For individual methods, we indicate in brackets [] the statistical coefficients that were used, and, depending on the situation, the width of the prediction channel and the application of Holt-exponential smoothing are also indicated. Each method is uniquely determined by its specific detection function:

DTV [V, c, ϱ]

- Calculation of moving coefficients of variation $V(t)$, autoregressive coefficient $c(t)$, and correlation coefficient $\varrho(t)$;
- Signaling of exceeding threshold values using the function $Y_\theta(t)$;

- Attack detection using the detection function $D_V(t)$:

$$D_V(t) = \mathrm{Y}_V(t) \cdot \mathrm{Y}_c(t) + \mathrm{Y}_V(t) \cdot \mathrm{Y}_\varrho(t) + \mathrm{Y}_c(t) \cdot \mathrm{Y}_\varrho(t)$$

DPT-3σ [V, K, Skw]

- Calculation of moving coefficients of variation $V(t)$, kurtosis $K(t)$, and skewness $Skw(t)$;
- Signaling of exceeding the upper limit 3σ-tunnel using the function $G_\theta(t)$;
- Attack detection using the detection function $D_T(t)$:

$$D_T(t) = G_V(t) \cdot G_K(t) + G_V(t) \cdot G_{Skw}(t) + G_K(t) \cdot G_{Skw}(t)$$

DPT-Hs-3σ [V, K, Skw]

- Calculation of moving coefficients of variation $V(t)$, kurtosis $K(t)$, and skewness $Skw(t)$;
- Coefficient values are smoothed using Holt-exponential smoothing;
- Signaling of exceeding the upper limit 3σ-tunnel using the function $H_\theta(t)$;
- Attack detection using the detection function $D_H(t)$:

$$D_H(t) = H_V(t) \cdot H_K(t) + H_V(t) \cdot H_{Skw} + H_K(t) \cdot H_{Skw}(t)$$

DPT-Hs-2σ [V, K, Skw, ϱ]

- Calculation of moving coefficients of variation $V(t)$, kurtosis $K(t)$, skewness $Skw(t)$, and correlation $\varrho(t)$;
- Coefficient values are smoothed using Holt-exponential smoothing;
- Signaling of exceeding the upper limit 2σ-tunnel using the function $H_\theta(t)$;
- Attack detection using multiplicative detection function $D_M(t)$:

$$D_M(t) = H_V(t) \cdot H_K(t) \cdot H_{Skw}(t) \cdot H_\varrho(t)$$

In the following Table 8, we present a summary evaluation of the use of detection functions on the measured records of DDoS attacks. As before, the first value represents the number of false reports during normal IP traffic. The second value determines the time (number of time slots) after which they started to detect the attack with minimal statistical coefficients (except for the function $D_M(.)$). The third value indicates the time when all three coefficients detected the attack. In the case of the $D_M(.)$ function, only two values are given because this function is a 0/1 value. The sign "x" means that the detection function did not recognize the attack in the given traffic capture.

Table 8. Summary evaluation of attack recognition using detection functions.

	S1	**N2**	**LR3**	**S5**	**N6**	**N7**	**N8**
$D_V(t)$	0/34/34	0/5/5	x	x	0/75/119	0/228/480	0/47/210
$D_T(t)$	1/6/28	0/1/1	1/1/1	1/18/41	1/12/12	1/39/49	0/1/1
$D_H(t)$	1/39/61	0/3/3	0/9/12	0/39/39	0/13/13	0/68/81	0/13/21
$D_M(t)$	1/51	0/7	0/43	1/41	$0/-106$	0/92	0/26

The given Table 4 provides a basic idea of the effectiveness of the proposed detection methods. However, it is not easy to make a single recommendation.

The computationally fastest method is DTV, which uses threshold values for detection. The method is effective for most standard DDoS attacks, but it could not detect non-standard attacks such as LR3 and S5 at all. Another disadvantage is the detection delay time, which is longer than with other methods.

With the DPT method, the detection delay time is several times shorter, even in some cases instantaneous, but it also has rare false positive reports.

With the DPT-Hs method, the computational difficulty is increased by using Holt-exponential smoothing. Smoothing mostly removed false reports, but extended the attack detection time.

The problematic traffic capture P4 was only handled by the DPT-Hs method using four statistical coefficients and reducing the width of the prediction tunnel to 2σ. The number of false reports caused by the reduced width was eliminated partly by Holt-exponential smoothing and partly by the multiplicative form of the detection function $D_M(t)$. Attack detection occurs only if all four coefficients $V(t)$, $K(t)$, $Skw(t)$, and $\varrho(t)$ recognize it. When testing traffic capture N6, a curious situation occurred when the detection function responded 106 ts earlier than the attack occurred. We can evaluate this event that the method detected the anomaly even before the start of the attack itself, which means, that the method totally failed in detecting the attack.

From the point of view of our subjective evaluation and also due to the calculation speed, we would still recommend the DPT-3σ method for hardware implementation, which uses the detection function $D_T(t)$ with coefficients of variation $V(t)$, kurtosis $K(t)$, and skewness $Skw(t)$.

9. Conclusions

In the article, we discussed the use of statistical coefficients for the quick detection of DDoS attacks. For various types of DDoS attack records, we calculated the moving coefficient of variation, kurtosis, skewnes, moving Hurst exponent, entropy, autoregressive and correlation coefficients, and moving Kullback–Leibler divergence. Based on their course, we divided the coefficients into two groups depending on whether we could use any of their threshold values or prediction methods when detecting an attack. For prediction, we created the predicting $n\sigma$-Tunnel, which uses the average coefficient and standard deviation value. For fast machine recognition of a DDoS attack, we have developed and tested four different methods, which are clearly determined by their own detection functions.

We used two parameters to determine the effectiveness of individual methods: the number of false reports during normal IP traffic and the delay time of DDoS attack detection.

During the development of detection methods, we gradually experimented with the size of the prediction tunnel, various combinations of statistical coefficients, exponential and Holt-exponential smoothing parameters, and different shapes of detection functions. The result is a custom recommendation for software implementation.

We partially excluded Hurst's exponent from considerations due to the computational complexity, variable course, and Entropy, which significantly reacted to peaks in normal traffic, which caused a lot of false reports.

We see great potential in the use of Kullback–Leibler divergence, which immediately reacted to the onset of a DDoS attack with a significant impulse. However, the problem was that this pulse was very short compared with the other coefficients, so we have not yet managed to include it in the detection functions.

We see the next direction of research in the search for types of detection functions that would use not yet tested statistical coefficients and, simultaneously, make machine recognition of DDoS attacks more efficient.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/math12010142/s1, Figure S1: Coefficient of Variation—Attack N2, Attack N6; Figure S2: Coefficient of Variation—Attack N7, Attack N8; Figure S3: Coefficient of Variation—Attack S1, Attack S5; Figure S4: Coefficient of Variation—Attack LR3, Attack P4; Figure S5: Kurtosis coefficient—Attack N2, Attack N6; Figure S6: Kurtosis coefficient—Attack N7, Attack N8; Figure S7: Kurtosis coefficient—Attack S1, Attack S5; Figure S8: Kurtosis coefficient—Attack LR3, Attack P4; Figure S9: Skewness coefficient—Attack N2, Attack N6; Figure S10: Skewness coefficient—Attack N7, Attack N8; Figure S11: Skewness coefficient—Attack S1, Attack S5; Figure S12: Skewness coefficient—Attack LR3, Attack P4; Figure S13: Entropy—Attack N2, Attack N6; Figure S14: Entropy—Attack N7, Attack N8; Figure S15: Entropy—Attack S1, Attack S5; Figure S16: Entropy—Attack LR3, Attack P4; Figure S17: Hurst exponent—Attack N2, Attack N6; Figure S18: Hurst

exponent—Attack N7, Attack N8; Figure S19: Hurst exponent—Attack S1, Attack S5; Figure S20: Hurst exponent—Attack LR3, Attack P4; Figure S21: Autoregression coefficient—Attack N2, Attack N6; Figure S22: Autoregression coefficient—Attack N7, Attack N8; Figure S23: Autoregression coefficient—Attack S1, Attack S5; Figure S24: Autoregression coefficient—Attack LR3, Attack P4; Figure S25: Correlation coefficient—Attack N2, Attack N6; Figure S26: Correlation coefficient—Attack N7, Attack N8; Figure S27: Correlation coefficient—Attack S1, Attack S5; Figure S28: Correlation coefficient—Attack LR3, Attack P4; Figure S29: Kullback-Leibler divergence—Attack N2, Attack N6; Figure S30: Kullback-Leibler divergence—Attack N7, Attack N8; Figure S31: Kullback-Leibler divergence—Attack S1, Attack S5; Figure S32: Kullback-Leibler divergence—Attack LR3, Attack P4.

Author Contributions: Conceptualization, J.S., P.S. and M.K.; methodology, J.S.; software, P.S.; validation, M.K.; formal analysis, J.S.; investigation, J.S. and M.K.; resources, M.K. and P.S.; data curation, M.K.; writing—original draft preparation, J.S.; writing—review and editing, P.S. and M.K.; visualization, J.S. and M.K.; supervision, J.S.; project administration, J.S.; funding acquisition, M.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the University of Žilina, Žilina, Slovakia grant scheme and by the Slovak Grant Agency VEGA project Fast Reroute, No. 1/0316/24.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data used in this study are freely available in the GIT repository: https://github.com/molcan23/RBC_NN (accessed on 10 November 2023).

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AR	Autoregression
DDoS	Distributed Denial of Service
GAN	Generative Adversarial Network
IP	Internet Protocol
MMRP	Markov-Modulated Regular Process
PCA	Principal Component Analysis
TCP	Transmission Control Protocol

References

1. Norton, N. What Is a DDoS and What Can You Do about Them? 2019. Available online: https://us.norton.com/blog/emerging-threats/what-is-a-ddos-attack-30sectech-by-norton (accessed on 10 November 2023).
2. Thottan, M.; Ji, C. Anomaly detection in IP networks. *IEEE Trans. Signal Process.* **2003**, *51*, 2191–2204. [CrossRef]
3. Bhattacharyya, D.K.; Kalita, J.K. *DDos Attacks: Evolution, Detection, Prevention, Reaction, and Tolerance*; CRC Press: Boca Raton, FL, USA, 2016; ISBN 978–1-4987-2964-2.
4. Lemeshko, O.; Papan, J.; Yeremenko, O.; Yevdokymenko, M.; Segeč, P. Research and Development of Delay-Sensitive Routing Tensor Model in IoT Core Networks. *Sensors* **2021**, *21*, 3934. [CrossRef] [PubMed]
5. Drozdova, M.; Bridova, I.; Uramova, J.; Moravcik, M. Private cloud security architecture. Paper presented at the ICETA 2020. In Proceedings of the 18th IEEE International Conference on Emerging eLearning Technologies and Applications, Košice, Slovenia, 12–13 November 2020; pp. 84–89. [CrossRef]
6. Hrabovsky, J.; Segeč, P.; Moravcik, M.; Papan, J. *Trends in Application of Machine Learning to Network-Based Intrusion Detection Systems*; Springer: Berlin/Heidelberg, Germany, 2018. [CrossRef]
7. Shiravi, A.; Shiravi, H.; Tavallaee, M.; Ghorbani, A.A. Toward developing a systematic approach to generate benchmark datasets for intrusion detection. *Comput. Secur.* **2012**, *31*, 357–374. [CrossRef]
8. Hajtmanek, R.; Kontšek, M.; Smieško, J.; Uramová, J. One-Parameter Statistical Methods to Recognize DDoS Attacks. *Symmetry* **2022**, *14*, 2388. [CrossRef]
9. Ye, N. *Secure Computer and Network Systems: Modeling, Analysis and Design, West Sussex*; John Wiley Sons Ltd.: Hoboken, NJ, USA, 2008; p. 336, ISBN 9780470023242. [CrossRef]
10. Fouladi, R.; Kayatas, C.; Anarim, E. Statistical measures: Promising features for time series based DDoS attack detection. *Proceedings* **2018**, *2*, 96. [CrossRef]

11. Erhan, D.; Anarim, E. Statistical Properties of DDoS Attacks. *Proceedings* **2018**, *2*, 96. [CrossRef]
12. Gupta, B.; Agawal, P.K.; Joshi, R.C.; Misra, M. Estimating Strength of a DDoS Attack Using Multiple Regression Analysis. 2012-03. Available online: https://www.inderscienceonline.com/doi/abs/10.1504/IJMIS.2010.039238 (accessed on 10 November 2023).
13. Gupta, B. Predicting Number of Zombies in DDoS Attacks Using Pace Regression Model. 2012-04. Available online: http://cit.fer.hr/index.php/CIT/article/view/1840 (accessed on 10 November 2023).
14. Chahar, N. Computer Network Security. *Int. J. Innov. Res. Sci. Eng. Technol.* **2022**, *7*, 1031. [CrossRef]
15. Huang, C.; Yi, P.; Zou, F.; Yao, Y.; Wang, W.; Zhu, T. CCID: Cross-Correlation Identity Distinction Method for Detecting Shrew DDoS. *Wirel. Commun. Mob. Comput.* **2019**, *2019*, 6705347. [CrossRef]
16. Rup, D.; Deka, K. Self-Similarity Based DDoS Attack Detection Using Hurst Parameter. Available online: https://onlinelibrary.wiley.com/doi/epdf/10.1002/sec.1639 (accessed on 10 November 2023).
17. Xia, H.; Xu, W. Research on Method of Network Abnormal Detection Based on Hurst Parameter Estimation. 2019. Available online: https://ieeexplore.ieee.org/document/4722405 (accessed on 10 November 2023).
18. Zheng, K.-F.; Wang, X.-J. Detecting DDoS attack with Hurst parameter of marginal spectrum. *Beijing Youdian Daxue Xuebao/J. Beijing Univ. Posts Telecommun.* **2011**, *34*, 128–132.
19. Li, M. Change trend of averaged Hurst parameter traffic under DDOS flood attacks. *Comput. Secur.* **2006**, *25*, 213–220. [CrossRef]
20. Dymora, P.; Mazurek, M. Network Anomaly Detection Based on the Statistical Self-similarity Factor. *Lect. Notes Electr. Eng.* **2015**, *324*, 271–287. [CrossRef]
21. Xia, Z.; Lu, S.; Tang, J. Note on Studying Change Point of LRD Traffic Based on Li's Detection of DDoS Flood Attacking. 2010-06. Available online: https://www.hindawi.com/journals/mpe/2010/962435/ (accessed on 10 November 2023).
22. Yan, R.; Xu, G.; Qin, X. Detect and Identify DDoS Attacks from Flash Crowd Based on Self-Similarity and Renyi Entropy. 2017. Available online: https://ieeexplore.ieee.org/document/8244075 (accessed on 10 November 2023).
23. Barsukov, I.; Bobreshov, A.M.; Riapolov, M.P. Fractal Analysis Based Detection of DoS/LDoS Network Attacks. 2019. Available online: https://ieeexplore.ieee.org/document/8867618 (accessed on 10 November 2023).
24. Kirichenko, L.; Radivilova, T.; Ageiev, D.; Bulakh, V. Classification Methods of Machine Learning to Detect DDoS Attacks. 2019-10. Available online: https://ieeexplore.ieee.org/document/8924406 (accessed on 10 November 2023).
25. Alzahrani, R.J. Security Analysis of DDoS Attacks Using Machine Learning Algorithms in Networks Traffic. *Electronics* **2021**, *10*, 2919. [CrossRef]
26. Shieh, C.-S.; Nguyen, T.-T.; Lin, W.-W.; Huang, Y.-L.; Horng, M.-F.; Lee, T.-F.; Miu, D. Detection of Adversarial DDoS Attacks Using Generative Adversarial Networks with Dual Discriminators. *Symmetry* **2022**, *14*, 66. [CrossRef]
27. Kopcan, J.; Skvarek, O.; Klimo, M. Anomaly detection using Autoencoders and Deep Convolution Generative Adversarial Networks. *Transp. Res. Procedia* **2021**, *55*, 1296–1303. [CrossRef]
28. Shieh, C.-S.; Nguyen, T.-T.; Lin, W.-W.; Lai, W.K.; Horng, M.-F.; Miu, D. Detection of Adversarial DDoS Attacks Using Symmetric Defense Generative Adversarial Networks. *Electronics* **2022**, *11*, 1977. [CrossRef]
29. Skvarek, O.; Klimo, M.; Kopcan, J. PCA Tail as the Anomaly Indicator. In Proceedings of the 2020 18th International Conference on Emerging eLearning Technologies and Applications (ICETA), Košice, Slovenia, 12–13 November 2020; pp. 613–619. [CrossRef]
30. Salaria, S.; Arora, S.; Goyal, N.; Goyal, P.; Sharma, S. Implementation and Analysis of an Improved PCA technique for DDoS Detection. In Proceedings of the 2020 IEEE 5th International Conference on Computing Communication and Automation (ICCCA), Greater Noida, India, 30–31 October 2020; pp. 280–285. [CrossRef]
31. Razian, M. TCP Low Rate DDoS Attack Detection. In Proceedings of the 3th International Conference on Applied Researches in Computer and Information Technology, Tehran, Iran, 4 February 2016.
32. Zhou, L.; Liao, M.; Yuan, C.; Haoyu, Z. Low-Rate DDoS Attack Detection Using Expectation of Packet Size. *Secur. Commun. Netw.* **2017**, *2017*, 3691629. [CrossRef]
33. Lysenko, S. Detection of the Botnets' Low-Rate DDoS Attacks Based on Self-Similarity. Available online: https://ijece.iaescore.com/index.php/IJECE/article/view/20780 (accessed on 10 November 2023).
34. Wei, W.; Song, H.; Wang, H.; Fan, X. Research and Simulation of Queue Management Algorithms in Ad Hoc Networks Under DDoS Attack. *IEEE Access* **2017**, *5*, 27810–27817. [CrossRef]
35. Xunyi, R. Wavelet analysis method for detection of DDoS attack on the basis of self-similarity, Frontiers of Electrical and Electronic Engineering in China. *March* **2007**, *2*, 73–77. [CrossRef]
36. Li, M.; Li, M. A New Approach for Detecting DDoS Attacks Based on Wavelet Analysis. 2009. Available online: https://ieeexplore.ieee.org/document/5300903 (accessed on 10 November 2023).
37. Barford, P.; Kline, J.; Plonka, D.; Ron, A. A signal analysis of network traffic anomalies. In Proceedings of the 2nd ACM SIGCOMM Workshop on Internet Measurment, Marseille, France, 6–8 November 2002; pp. 71–82.
38. Wani, S.; Imthiyas, M.; Almohamedh, H.; Alhamed, K.M.; Almotairi, S.; Gulzar, Y. Distributed Denial of Service (DDoS) Mitigation Using Blockchain—A Comprehensive Insight. *Symmetry* **2021**, *13*, 227. [CrossRef]
39. Zhu, P.; Hu, J.; Li, X.; Zhu, Q.; Using Blockchain Technology to Enhance the Traceability of Original Achievements. *IEEE Trans. Eng. Manag.* **2023**, *70*, 1693–1707. [CrossRef]
40. Alduailij, M.; Khan, Q.W.; Tahir, M.; Sardaraz, M.; Alduailij, M.; Malik, F. Machine-Learning-Based DDoS Attack Detection Using Mutual Information and Random Forest Feature Importance Method. *Symmetry* **2022**, *14*, 1095. [CrossRef]

41. Javadpour, A.; Ja'fari, F.; Taleb, T.; Shojafar, M.; Yang, B.; SCEMA: An SDN-Oriented Cost-Effective Edge-Based MTD Approach. *IEEE Trans. Inf. Forensics Secur.* **2023**, *18*, 667–668. [CrossRef]
42. Gelenbe, E.; Pujolle, G. *Introduction to Queueing Networks*; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 1987.
43. Walrand, J. *An Introduction to Queueing Networks*; Prentice Hall: Englewood Cliffs, NJ, USA, 1988.
44. Kelly, F.P. *Notes on Effective Bandwidth, Stochastic Networks: Theory and Application*; Oxford University Press: Oxford, UK, 1996; pp. 141–168.
45. Chang, C.S. *Performance Guarantees in Communication Networks*; Springer: Berlin/Heidelberg, Germany, 2000.
46. Le Boudec, J.Y.; Thiran, P. *Network Calculus: A Theory of Deterministic Queuing Systems for the Internet*; Springer: Berlin/Heidelberg, Germany, 2001. [CrossRef]
47. Uramova, J. Infrastructure for Generating New IDS Dataset. In Proceedings of the 2018 16th International Conference on Emerging eLearning Technologies and Applications (ICETA), Starý Smokovec, Slovakia, 15–16 November 2018.
48. Sharafaldin, I.; Gharib, A.; Lashkari, A.H.; Ghorbani, A.A. Towards a Reliable Intrusion Detection Benchmark Dataset. *Softw. Netw.* **2017**, *2017*, 177–200. [CrossRef]
49. Sharafaldin, I.; Lashkari, A.H.; Ghorbani, A.A. Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. *ICISSp* **2018**, *1*, 108–116.
50. Lei, G.; Ji, L.; Ji, R.; Cao, Y.; Shao, X.; Huang, X. Extracting Low-Rate DDoS Attack Characteristics: The Case of Multipath TCP-Based Communication Networks. *Wirel. Commun. Mob. Comput.* **2021**, *2021*, 2264187. [CrossRef]
51. Alashhab, A.A.; Zahid, M.S.M.; Azim, M.A.; Daha, M.Y.; Isyaku, B.; Ali, S. A Survey of Low Rate DDoS Detection Techniques Based on Machine Learning in Software-Defined Networks. *Symmetry* **2022**, *14*, 1563. [CrossRef]
52. Santos, C.; Dias, C. Note on the coefficient of variation properties. *Braz. Electron. J. Math.* **2021**, *2*, 101–111. [CrossRef]
53. Smiesko, J.; Uramova, J. One-parameter Methods for Recognizing DDoS Attacks ICETA 2020. In Proceedings of the 18th IEEE International Conference on Emerging eLearning Technologies and Applications, Košice, Slovenia, 12–13 November 2020; pp. 628–633.
54. Sheng, H.; Chen, Y.Q.; Qiu, T. On the robustness of Hurst estimators. *IET Signal Process.* **2011**, *5*, 209–225. [CrossRef]
55. Lenskiy, A.; Seol, S.; The Analysis of R/S Estimation Algorithm with Applications to WiMAX Network Traffic. *Int. J. Multimed. Ubiquitous Eng.* **2012**, *7*, 27–34.
56. Zournatzidou, G.; Floros, C. Hurst Exponent Analysis: Evidence from Volatility Indices and the Volatility of Volatility Indices. *J. Risk Financ. Manag.* **2023**, *16*, 272. [CrossRef]
57. Mariani, M.C.; Kubin, W.; Asante, P.K.; Guthrie, J.A.; Tweneboah, O.K. Relationship between Continuum of Hurst Exponents of Noise-like Time Series and the Cantor Set. *Entropy* **2021**, *23*, 1505. [CrossRef] [PubMed]
58. Ambriško, R. Aplikácia Teórie Chaosu na Menovom Trhu SR. 22 Apríl 2003. Available online: https://kipdf.com/ekonomicka-univerzita-v-bratislave-diplomova\praca_5aff16898ead0e19668b465b.html (accessed on 10 November 2023).
59. Feller, W. *An Introduction to Probability Theory and Its Applications*, 2nd ed.; John Wiley & Sons Inc.: New York, NY, USA, 1971; Volume II.
60. Montgomery, D.; Runger, G. *Applied Statistics and Probability for Engineers*; Wiley: Hoboken, NJ, USA, 2014; ISBN 978-1-118-74412-3
61. Halušková, E.P. Detegovanie IP úTokov Pomocou Predikcie časových Radov, Žilina. 2023. Available online: https://opac.crzp.sk/?fn=detailBiblioForm&sid=30EAE6FD1E2AFEEA16C1A31C8F40 (accessed on 10 November 2023).
62. Tripathy, B.K.; Anveshrithaa, S.; Ghela, S. Principal Component Analysis (PCA). In *Unsupervised Learning Approaches for Dimensionality Reduction and Data Visualization*, 1st ed.; CRC Press: Boca Raton, FL, USA, 2021. [CrossRef]
63. Brown, R. Exponential Smoothing. In *Encyclopedia of Operations Research and Management Science*; Gass, S.I., Fu, M.C., Eds.; Springer: Boston, MA, USA, 2013. [CrossRef]
64. Fried, R.; George, A.C. Exponential and Holt-Winters Smoothing. In *International Encyclopedia of Statistical Science*; Lovric, M., Ed.; Springer: Berlin/Heidelberg, Germany, 2011. [CrossRef]
65. Smieško, J.; Tomancová, S. Use of PCA Method for DDoS attack detection. *Elektrorevue* **2017**, *16*, 104–110.