Theoretical and Applied Aspects of Cybersecurity
of Convergent Cognitive Information Technologies

https://doi.org/10.25559/SITITO.021.202502.230-240
UDC 519.246.85

Statistical Analysis of Time Series for Portscan and DDoS Detection: Study of Factors Influencing Z-score Based Sliding Windows Algorithm

A. M. A. E. Djeguede

Peoples' Friendship University of Russia named after Patrice Lumumba, Moscow, Russian Federation

Address: 6 Miklukho-Maklaya St., Moscow 117198, Russian Federation

djeguede.marc@gmail.com

Original article

Abstract

In the course of this study, statistical methods for time series analysis – specifically, the Z-score and the modified Z-score – were investigated for the detection of PortScan and DDoS attacks. Six time series were constructed based on the following traffic features: the average number of packets transmitted from sources to destinations, the data transfer rate from source to destination, the response data transfer rate, the connection duration between the source and the destination, the entropy calculated based on the destination ports of each source IP, and the number of unique destination ports accessed by each source IP. To evaluate the aforementioned statistical methods, the metrics of accuracy, precision, recall, and F1-score were used. The numerical results show that the modified Z-score yields fewer false positives compared to the standard Z-score in detecting the studied network threats, which influences the evaluation of these metrics. The F1-scores achieved by the modified Z-score for detecting DDoS attacks range between 93% and 98%, depending on the traffic feature used. However, the F1-score for detecting PortScan attacks does not exceed 58% at best. A detailed analysis showed that all detected PortScan instances correspond to fast port scanning, as this type of scanning causes a spike in traffic. This effect is reflected in the local violation of the stationarity of the time series. These conclusions were confirmed by ADF and KPSS statistical tests, which were conducted to test different hypotheses regarding the stationarity of the series.

Keywords: time series analysis, anomaly detection, PortScan, DDoS, Z-score

Conflict of interests: The author declares no conflict of interests.

For citation: Djeguede A.M.A.E. Statistical Analysis of Time Series for Portscan and DDoS Detection: Study of Factors Influencing Z-score Based Sliding Windows Algorithm. *Modern Information Technologies and IT-Education.* 2025;21(2):230-240. https://doi.org/10.25559/SITITO.021.202502.230-240

Теоретические и прикладные аспекты кибербезопасности конвергентных когнитивно-информационных технологий

Статистический анализ временных рядов для обнаружения PortScan и DDoS: изучение факторов, влияющих на эффективность алгоритма скользящих окон на основе Z-оценки

А. М. А. Э. Джегюеде

ФГАОУ ВО «Российский университет дружбы народов имени Патриса Лумумбы», г. Москва, Российская Федерация

Адрес: 117198, Российская Федерация, г. Москва, ул. Миклухо-Маклая, д. 6

djeguede.marc@gmail.com

Оригинальная статья

Аннотация

В ходе выполнения работы в рамках этой статьи были исследованы статистические методы обработки временных рядов Z-оценка и модифицированная Z-оценка при обнаружении PortScan и DDoS аттак. Составлены 6 временных рядов из следующих характеристик трафика: среднее количество пакетов, переданных от источников к получателям, скорость передачи данных от источника к приемнику, скорость передачи данных ответа, продолжительность соединения между источником и пунктом назначения, вычисленная энтропия на основе портов назначения каждого источника IP, количество уникальных портов назначения, подключенных каждым источником IP. Для оценки выше перечисленных статистических методов были использованы метрики достоверности, точности, отклика и F1-меры. Полученные численные результаты показывают, что модифицированная Z-оценка вызывает меньше ложно-положительное срабатывание по сравнению с Z-оценкой при обнаружении изучаемых сетевых угроз, что влияет на оценку перечисленных метрик. Достигнутые показатели F1-меры модифицированной Z-оценкой при обнаружении DDoS атак в районе 93%-98% в зависимости от используемой характеристики трафика. F1-мера при обнаружении Portscan не превышает 58% в лучшем случае. Детальный анализ показал, что все обнаруженные PortScan относят к быстрому сканированию портов, так как данный вид сканирование вызывает всплеск трафика. Этот факт отражается через локальное нарушение стационарность временных рядов. Данные выводы подтверждены ADF и KPSS статистическими тестами, проведенными для проверки разных гипотез о стационарности рядов.

Ключевые слова: анализ временных рядов, обнаружение аномалий, PortScan, DDoS, Z-score

Конфликт интересов: автор заявляет об отсутствии конфликта интересов.

Для цитирования: Джегюеде А. М. А. Э. Статистический анализ временных рядов для обнаружения PortScan и DDoS: изучение факторов, влияющих на эффективность алгоритма скользящих окон на основе Z-оценки // Современные информационные технологии и ИТ-образование. 2025. Т. 21, № 2. С. 230-240. https://doi.org/10.25559/SITITO.021.202502.230-240

Introduction

Intrusion Detection Systems (IDS) are essential for protecting computer networks, as they monitor and analyze network traffic to detect possible security threats. As network data becomes increasingly complex and voluminous, traditional static analysis methods often struggle to recognize advanced or evolving attacks. Time series analysis provides a more dynamic solution by examining how network behavior changes over time. By interpreting metrics like connection frequency, packet flow, or data volume as time-based data, IDS can uncover subtle irregularities or patterns that may signal malicious activity. This time-aware approach improves the detection of both sudden and gradual threats, enabling the development of more intelligent and adaptable intrusion detection systems.

Port scanning represents a prevalent reconnaissance methodology employed by adversaries to ascertain accessible ports and services on a designated system. The identification of such activities is paramount for the preservation of network security, as they frequently precede more severe intrusions. Conventional detection methodologies typically depend on rule-based frameworks or threshold triggers, which may overlook nuanced or gradual scans engineered to circumvent detection. Time series analysis presents a robust alternative by scrutinizing patterns and anomalies in network traffic across temporal dimensions. By conceptualizing connection attempts as temporal datasets, it becomes feasible to identify deviations from normative behavior that may signify scanning activities, even in their more covert manifestations. This paradigm not only enhances detection precision but also facilitates proactive threat mitigation through the implementation of early warning systems.

Distributed Denial-of-Service (DDoS) attacks constitute a continual menace to digital services, with the objective of inundating targeted infrastructures (servers, networks, applications) with extensive, malevolent traffic, thus rendering them inaccessible to authorized users. Conventional signature-based detection methods frequently encounter difficulties when confronted with novel or rapidly adapting attack methodologies. Time Series Analysis (TSA) presents a data-centric alternative by concentrating on the intrinsic temporal patterns of network traffic to detect nuanced and significant anomalies that may signify an impending attack.

Materials

The CIC-IDS-2017 dataset is one of the most widely used cybersecurity datasets for evaluating Intrusion Detection Systems (IDS). The Canadian Institute for Cybersecurity (CIC) created it at the University of New Brunswick (UNB). It contains a comprehensive and realistic set of network traffic data, including both benign and malicious activities, modeled to reflect modern network behavior. The data was captured using realistic user behavior and attack scenarios, including modern threats like botnets and web attacks.

Each day includes specific attack types:

T a b l e 1. **CIC-IDS2017 traffic daily classification**

Day	Type of Traffic	Attack Types Present
Monday	Normal traffic only	None
Tuesday	Brute Force attacks	SSH Brute Force, FTP Brute Force
Wednesday	DoS and Heartbleed	DoS Hulk, DoS GoldenEye, Heartbleed
Thursday	Web and Infiltration	Web Attack (XSS, SQLi, Cmd Injection), Infiltration
Friday	Botnet and PortScan	Botnet, Port Scan, DDoS

Source: Hereinafter in this article all tables and figures were made by the author.

In this work, we will explore time series analysis techniques to detect PortScan attacks and DDoS attacks. In port scanning, attackers can collect information about the port number, OS, and applications used by the host by sending specific data to the host port on the network and analyzing the response. For example, a simple port scan can be performed by a method called TCP scan, which checks if a port is in use by trying the 3-Way Handshake on the target port. However, since the 3-Way Handshake is performed even in standard TCP connections, it is difficult to distinguish between a port scan attack and regular communication on a packet-by-packet basis. Therefore, it is necessary to find features from multiple packets in order to identify port scans, such as the amount of port accesses per unit of time from the same host to different hosts or the amount of traffic per port .There are many behaviors characterizing a scanning activity such as:

- **Numerous Connection Attempts**: A scanning mechanism dispatches a multitude of connection requests directed towards various ports on a designated machine.
- **Sequential or Patterned Approach**: Ports are typically examined in a sequential manner (for instance, 1, 2, 3, 4…) or according to a predetermined pattern (prioritizing common ports, e.g., 22, 80, 443).
- **Transient Connections**: Connections are frequently of a fleeting nature – sufficiently brief to ascertain whether the port is open, closed, or filtered.
- **Anomalous Traffic Volume**: Relative to normative behavior, port scanning engenders an elevated number of connection attempts, frequently within a condensed period.
- **Unexpected Origin**: Scanning activities predominantly originate from external or unrecognized IP addresses.

Beside the above-described scanning characteristics, the attackers can employ more specific scanning techniques:

- **SYN scan (half-open):** Transmits SYN packets while abstaining from completing the TCP handshake.
- **FIN, XMAS, NULL scans:** Dispatch unconventional TCP flags to circumvent firewalls or to discreetly identify ports.
- **UDP scan**: Probes UDP ports, though slower because of lack of clear feedback.

Depending on variations in scanning timing, we have:

- **Aggressive** (rapid scan = heightened likelihood of detection),

- **Stealthy** (deliberate scan over extended periods to mitigate the risk of detection).

According to these characteristics in huge amount of research works, time series for the following features were computed and analyzed:

- **Connection Count per Time Window (Burstiness)** – Number of connection attempts from a source IP within a short time window.
- **Unique Destination Ports Count** – Count of distinct destination ports accessed by a single IP in a given timeframe.
- **Entropy of Destination Ports** – Entropy of the destination port numbers targeted by a source IP over time.
- **Packet Size and Time Interval Patterns** – Regular intervals and uniform packet sizes are typical of automated scanners.
- **Ratio of Failed to Successful Connections** – Scans usually trigger connection failures for instance RST flags in TCP.

Another threat we will study in this work is distributed denial of service. DDoS attacks remain a persistent threat in cybersecurity, evolving in scale and sophistication. Proactive defense strategies, combined with rapid incident response plans, are essential for minimizing their impact. Unlike a traditional Denial-of-Service (DoS) attack, which originates from a single source, a DDoS attack leverages a distributed network of compromised devices (a botnet) to amplify its impact, making mitigation more challenging. DDoS has a variety of attack vectors such as:

- **Volume-Based Attacks** – Flood the target with high traffic volumes to exhaust bandwidth for example UDP/ICMP Floods, that consists in sending spoofed UDP or ICMP packets and amplification attacks exploiting protocols DNS, NTP to magnify traffic by triggering large responses to small requests.
- **Protocol Attacks** – Overwhelm the TCP handshake process, leaving connections half-open (SYN Floods) or send malformed packets to crash systems (Ping of Death).
- **Application-Layer Attacks** – Target specific applications (e.g., web servers) with resource-intensive requests, mimicing legitimate user traffic to overload servers (HTTP Floods e.g., repeated page requests) or keeping server connections open indefinitely to exhaust resources (Slowloris).

Methods

Effective anomaly detection is key to ensuring the security and stability of network infrastructure. One approach to detecting deviations in network behavior is time series analysis. A time series is a sequence of observed values ordered by time. In the context of networks, examples of time series are:

• umber of incoming/outgoing packets per second;
• average latency per interval;
• channel load level;
• number of connections to a particular service.

The following features were extracted from the previous datasets, as described in Table 1.

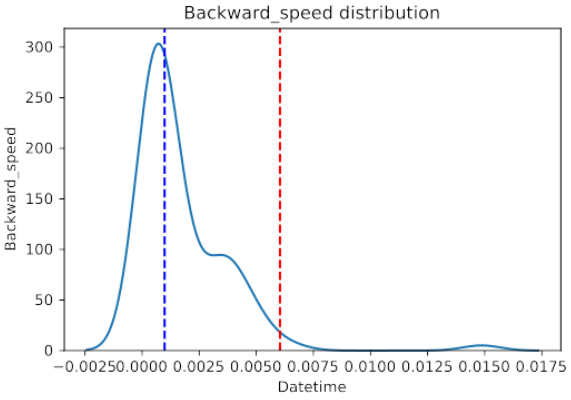
To analyze these time series and detect anomalies in them, we will use statistical Z-score calculation for each *i*-point using rolling window w_i with size d. Formula of Z-score calculation can be described as $Z_i := \frac{x_i - \mu(w_i)}{\sigma_i(w_i)}$, where the local mean $\mu(w_i) = \frac{1}{d} \cdot \sum_{j=i-d}^{i-1} x_j$ and $\sigma(w_i) = \sqrt{\frac{1}{d} \cdot \sum_{j=i-d}^{i-1} [x_j - \mu(w_i)]^2}$ is the standard deviation. The main assumption for using Z-score is the normal distribution of studied features.

T a b l e 2. **Selected features for DDoS detection**

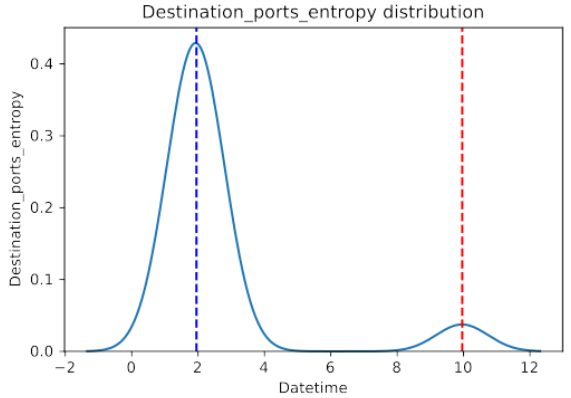
Features	Descriptions
Mean Forward packets per IP source	Average number of packets transmitted from sources to receivers
Forward speed	Data transfer rate from source to receiver
Backward speed	Response data transfer rate

T a b l e 3. **Selected features for PortScan detection**

Features	Descriptions
Connection live time	Duration of connection between source and destination
Destination ports entropy per source IP	Computed entropy based on destination ports of each IP source
Unique destination ports count by source IP	Count of unique destination ports connected by each IP source



c. Backward speed distr



f. Destination ports entropy distr

F i g. 1. Feature distribution functions

An empirical study of the distribution of these features is shown in the graphs Fig. 1. These distributions are near to Gaussian distribution, and then the decision to use the Z-score in this work is justified. In the Table 4 there is a sample algorithm based on Z-score to detect outliers in time series.

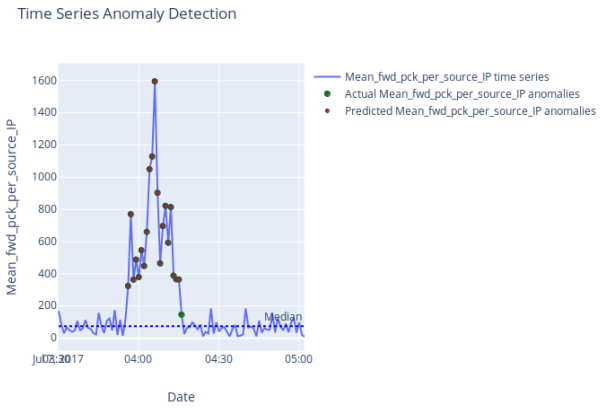
T a b l e 4. **Detection of anomalies in time series**

Algorithm 1 Detection of anomalies in time series
1: $T = \{x_1, \cdots, x_n\}$ – time series
2: d – size of window
3: t – threshold
4: **for** $i \in \{1, \cdots, n\}$ **do**
5: select window $w_i = \{x_{i-d}, x_{i-d+1}, \dots, x_{i-1}\}$
6: compute z-score $Z_i := \frac{x_i - \mu(w_i)}{\sigma(w_i)}$
7: **if** $\lvert Z_i \rvert \geq t$ **then**
8: labelize x_i as anomaly
9: **else**
10: labelize $x_i$as benign
11: **end if**
12: **end for**

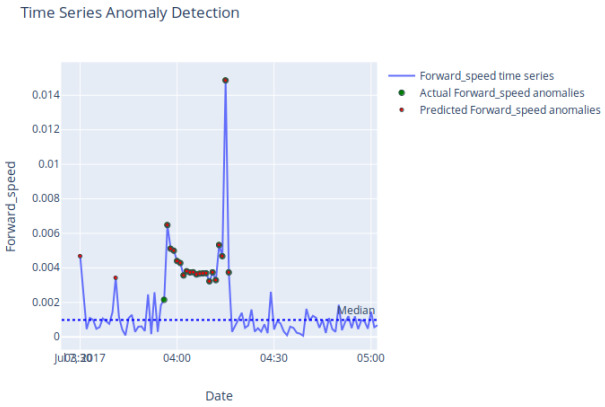
For several reasons that will be explained in the next section, the algorithm using the z-score metric calculation and sliding windows can be inefficient. In this case, we will use modified z score. Modified Z-score can be computed by $Z_{modified_{score}} = 0.6745 * \frac{x_i - median}{mad}$, where mad – stands for median absolute deviation and can be calculated by $mad = \sqrt{\frac{1}{N} \cdot \sum_{j=i-N}^{i-1} (x_j - median)^2}$.

Results and Discussions

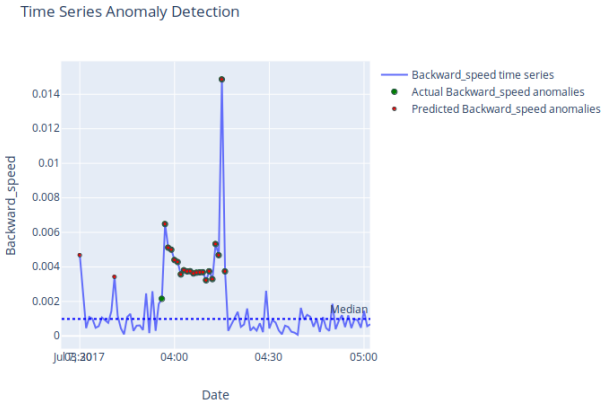
This section is dedicated to present the results of experimentation. In Fig. 2 and we depicted the time series obtained from features, that we described in Table 2 for DDoS whereas in Fig. 3 we has the representation of time series obtained from features in Table 3 for PortScan detection. In these figures green points on time series stand for actual threats (DDoS and PortScan), whereas red points for predicted threats. Visual analysis of these results shows that modified Z-score performs better, than sliding algorithm depicted in Table 4.



b. Modified Z-score with mean forward packets

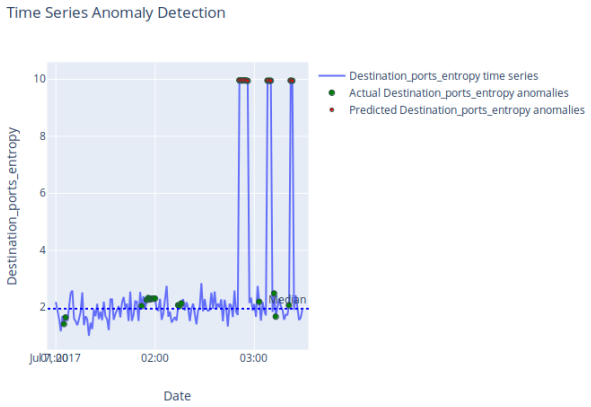


d. Modified Z_score with forward speed time series



f. Modified Z-score with backward speed time series

F i g. 2. DDoS attacks detection in time series using Z-score and Modified Z-score



f. Modified Z-score with destination ports entropy

F i g. 3. PortScan detection in time series using Z-score and Modified Z-score

For more detailed assessment of performance of two methods we will use confusion matrix with structure explained in Table 5 to compute metrics – accuracy, precision, recall, f1 using but using following formulas $Accuracy = \frac{TruePositives + TrueNegatives}{TotalPredictions}$,

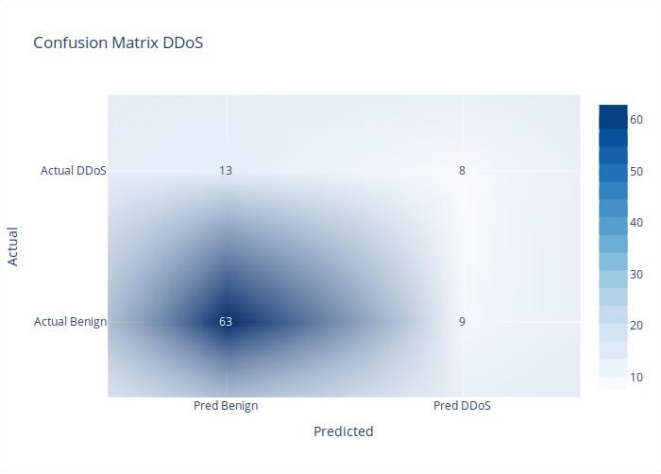
$Precision = \frac{TruePositives}{TruePositives + FalsePositives}$,

$Recall = \frac{TruePositives}{TruePositives + FalseNegatives}$, and

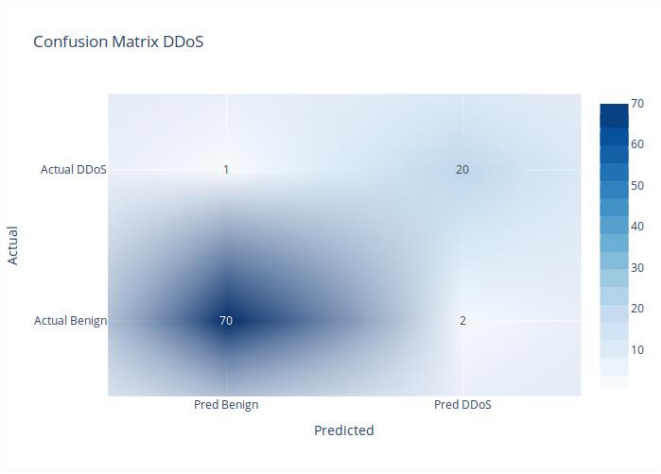
$F1 = 2 * \frac{Recall * Precision}{Recall + Precision}$

T a b l e 5. **Confusion Matrix Structure**

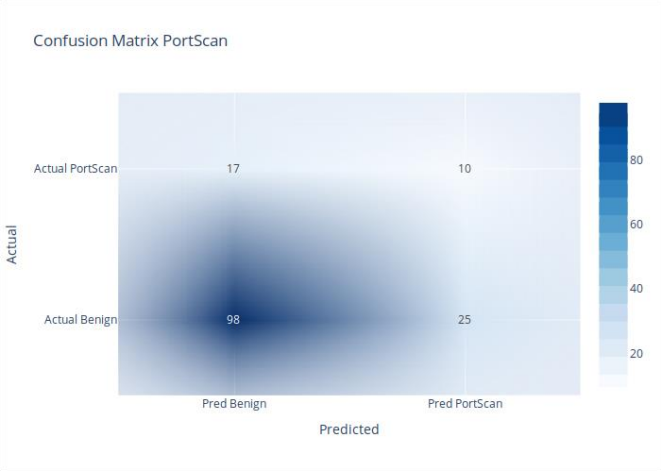
	Predicted Positive	**Predicted Negative**
Actual Positive	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)



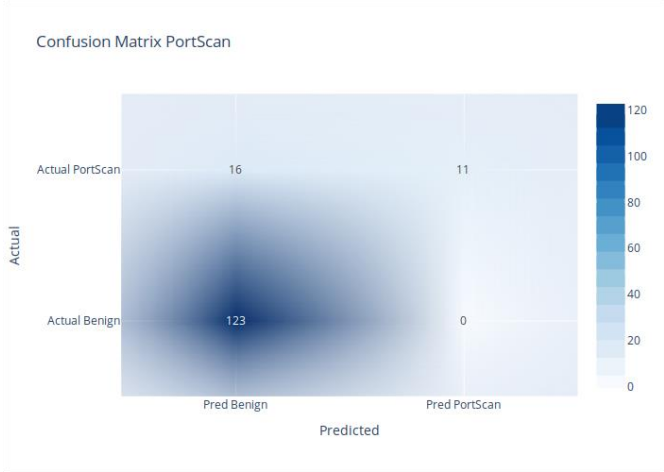
c. Z-score with backward speed



f. Modified Z-score with backward speed
F i g. 4. Confusion matrix for DDoS attack detection



c. Z-score with destination ports entropy



f. Modified Z-score with destination ports entropy
F i g. 5. Confusion matrix for PortScan detection

Based on confusion matrix drawn in Fig. 4 and Fig. 5 we can compute the metrics reported in Table 6 and Table 7. According to these results, we can conclude:

- low accuracy of the assessment indicates a higher number of false positives,
- low recall indicates a higher number of false negatives.

Using sliding windows algorithm with Z-score, we obtained low accuracies and recalls for detection both DDoS and PortScan (generally under 50%) for all selected features, whereas global modified Z-score perfoms very well for DDoS detection (accuracy, recall and F1 generally over 91%). The performance of the global Z-score is more debatable in the context of PortScan detection.

Main hypothesis justifying the power performance of sliding algorithm with Z-score is non-stationary behavior of time series. For the sake of corroborate this hypothesis we compute the p-value of ADF and KPSS statistics in Table 8. We have also studied the behavior of statistical properties (mean and standard deviation) of time series over the time in Fig. 6.

For a better understanding of performance of modified Z-score while detecting port scanning activities, it is important to look at Fig. 3 b, d, f. It can be seen that all detected threats have a high variance, while threats with a small variance bypass detection. If we compare the time intervals of the appearance of threats with high dispersion, with the data from the dataset before their regrouping and taking into account the number of flows with label Portscan, we can say that we are talking about fast Portscan. From all this, we can conclude that the modified Z -score method copes with the detection of fast port scanning, while it cannot cope with slow port scanning.

This part is devoted to study the influence of local non-stationarity of feature time series in detection of port scanning activities. For this sake, we compute the p-value of ADF and KPSS statistics. We can express p-value formula as $p_{value} = P(x_{obs}|H_0)$ where H_0- null hypothesis, x_{obs}- is the calculated statistic (ADF or KPSS). The general form of the ADF test regression is:

$$\Delta y_t = \alpha + \beta t + \gamma y_{t-1} + \sum_{i=1}^{p} \delta_i \Delta_{t-i} + \epsilon_t$$

T a b l e 6. **DDoS detection metrics (Accuracy, Precision, Recall and F1)**

			Accuracy	Precision	Recall	F1	Support
Mean forward packets by source IP	**Z-score**	BENIGN	0.75	0.82	0.88	0.85	72
		DDoS		0.44	0.33	0.38	21
	Modified Z-score	BENIGN	0.99	0.99	1	0.99	72
		DDoS		1	0.95	0.98	21
Forward speed	**Z-score**	BENIGN	0.76	0.83	0.88	0.85	72
		DDoS		0.47	0.38	0.42	21
	Mofdiied Z-score	BENIGN	0.97	0.99	0.97	0.98	72
		DDoS		0.91	0.95	0.93	21
Backward speed	**Z-score**	BENIGN	0.76	0.83	0.88	0.85	72
		DDoS		0.47	0.38	0.42	21
	Modified Z-score	BENIGN	0.97	0.99	0.97	0.98	72
		DDoS		0.91	0.95	0.93	21

T a b l e 7. **PortScan detection metrics (Accuracy, Precision, Recall and F1)**

			Accuracy	Precision	Recall	F1	Support
Connection live time	**Z-score**	BENIGN	0.813	0.89	0.89	0.89	123
		PortScan		0.48	0.48	0.48	27
	Modified Z-score	BENIGN	0.860	0.88	0.96	0.92	123
		PortScan		0.69	0.41	0.51	27
Unique destination count by source IP	**Z-score**	BENIGN	0.76	0.87	0.84	0.85	123
		PortScan		0.35	0.41	0.38	27
	Mofdiied Z-score	BENIGN	0.867	0.88	0.97	0.92	123
		PortScan		0.73	0.41	0.52	27
Destination port entropy	**Z-score**	BENIGN	0.72	0.85	0.80	0.82	123
		PortScan		0.29	0.37	0.32	27
	Modified Z-score	BENIGN	0.893	0.88	1.00	0.94	123
		PortScan		1.00	0.41	0.58	27

- Null hypothesis (H_0): $\gamma = 0 \rightarrow$*unit root is present (non-stationary).*
- Alternative hypothesis (H_1): $\gamma < 0 \rightarrow$*no unit root (stationary).*

Then we can compute $ADF_{stat} = \frac{\hat{\gamma}}{SE(\gamma)}$. As p_{value}*is more than 0.05*, so we fail to reject null hypothesis.

In the next step, we estimate KPSS Test statistic formula as $KPSS = \frac{1}{T^2}\frac{\sum_{t=1}^{T} S_t^2}{\hat{\sigma}^2}$ *where:*

- T: Number of observations.
- $S_t = \sum_{i=1}^{t} \hat{e}_i$ *: The* cumulative residuals *from the OLS regression of the time series on a constant (or constant + trend).*
- $\hat{\sigma}^2$*:* ***Long-run variance*** *of the residuals, estimated using a* Newey-West estimator *(or similar), which accounts for autocorrelation.*

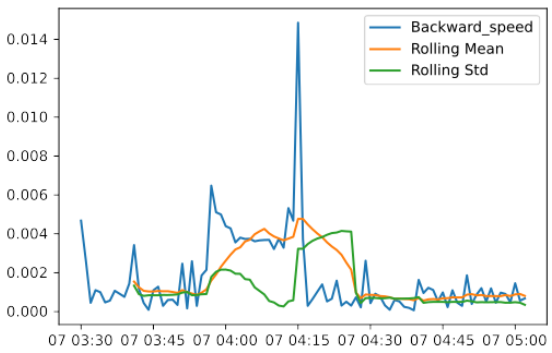
Null Hypothesis (H_0): *The series is* stationary *(level or trend stationary).*

Alternative Hypothesis (H_1): *The series has a* unit root *(non-stationary).*

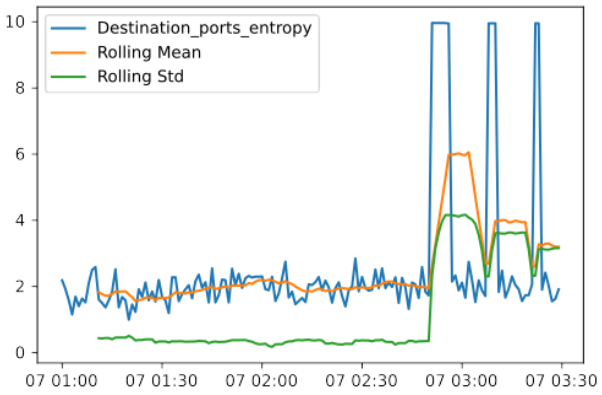
We reject H_0 (stationary) if the $p_{value} < 0.05$ (KPSS statistic is greater than the critical value).

T a b l e 8. **ADF and KPSS statistics and p-values of time series**

	ADF		KPSS	
	ADF Statistic	**p-value**	**Kpss Statistic**	**p-value**
Mean forward packets	-2.68053	0.0774668	0.279649	0.1
Forward speed	-3.6409978	0.0050239	0.3529392	0.0974400
Backward speed	-3.6409978	0.0050239	0.3529392	0.0974400
Connection live time	-4.8111134	5.1785264e-05	0.1351005	0.1
Unique destination ports count	-1.9973058	0.2877557	0.5839902	0.0240918
Destination ports entropy	-1.9618836	0.3035317	0.6503301	0.0180609



c. Backward speed



f. Destination ports entropy

F i g. 6. Rolling statistical characteristics for feature time series

Analyzing p-value Table 8 of KPSS test of time series used in port scanning detection we can conclude that 'Connection live' time series is stationary, whereas 'Unique destination ports count' and 'destination ports entropy' time series are not stationary. Non-stationary state of these time series is caused by the pick of traffic due to fast port scanning activities according to Fig. 6 d, e, f, therefore high KPSS statistic of local stationary analysis can be a good indication of period of fast port scanning activities.

Literature review

The centralized architecture of software-defined networks (SDN) makes them ideal targets for flood attacks such as Distributed Denial of Service (DDoS) and port scanning. Addressing this issue, G. F. Scaranti et al. [1] proposed an intrusion detection system (IDS) based on online clustering to detect attacks in evolving SDN networks by leveraging the entropy of source and destination IP addresses and ports. The proposed system eliminates the need for data labeling, paving the way for comprehensive analysis by projecting cluster structures into the feature space and providing information on the intensity, seasonality, and type of various attacks. The system is built on the DenStream algorithm [1], utilizing multiple databases targeted by DDoS and port scanning attacks with varying intensity and duration [2].

C. Birkinshaw et al. in [3] also designed a Software-Defined Networking (SDN)-based Intrusion Detection and Prevention System (IDPS). The proposed system is a software application that monitors the network for malicious activity or security policy violations and takes measures to mitigate such activity. Special emphasis is placed on protecting against port scanning and Denial-of-Service (DoS) attacks. During their work, the authors described and tested the Port Bingo (PB) algorithm as a defense mechanism against port scanning, and implemented two connection-based methods: Credit-Based Threshold Random Walk (CB-TRW) and Rate Limiting (RL) [4].

Botnets are responsible for some of the most significant malicious attacks on the internet: DDoS attacks, email spam, brute-force attacks, port scanning, and others. Their danger stems from the coordinated actions of infected hosts targeting a single objective. R. Abrantes et al. in their article [5] focused on identifying botnet traffic to prevent

communication between the botmaster and infected hosts. For analyzing hosts in the Botnet2014 dataset, they employed the CICFlowMeter algorithm and machine learning methods – Random Forest (RF) and Decision Tree (CART). The results show that the analysis scenario using IP addresses and L4 ports achieves higher accuracy but a lower F1-score compared to equivalent scenarios without IP addresses or L4 ports [6].

Detecting low-frequency attacks requires additional computational overhead compared to regular traffic. In [7], D. Ono et al. proposed a method for detecting port scanning by analyzing the characteristics of Packet-In messages sent from an OpenFlow (OF) switch to the controller. Port scanning typically generates a large volume of Packet-In messages. The proposed method monitors the flow rate of Packet-In messages sent by each host to the switch, identified by their addresses. Upon detecting an abnormal increase in this rate, the controller requests statistics from the switch and implements an algorithm based on the collected data to identify port scanning. The employed algorithm significantly reduces computational costs compared to conventional methods [8].

Port scanning, commonly used as a reconnaissance tool in attacks, can create significant performance and bandwidth challenges for applications. In [9] B. Hartpence et al. describe an architecture of recurrent neural networks (RNNs) for packet classification, TCP datagram separation, TCP packet type identification, and port scanning detection. Recurrent neural networks enable learning temporal dependencies in prolonged port scanning sequences that unfold over time. Testing the proposed model in this work with real NMAP application pcap files demonstrated successful detection of open ports and scanning attempts with high accuracy and a low false-positive rate [10].

Q. Abu Al-Haija et al. in [11] proposed a novel inclusive port scanning detection scheme that evaluates five machine learning classifiers, including logistic regression, decision trees, linear/quadratic discriminant analysis, naive Bayes, and ensemble boosted trees. Studies conducted on the modern PSA-2017 dataset demonstrated the best performance for the logistic regression-based detection scheme, achieving 99.4% accuracy, 99.9% precision, 99.4% recall, 99.7% F-score, and a detection time of 0.454 μsec.

SNORT is an intrusion detection and prevention system (IDS/IPS) and a popular tool for monitoring network traffic in real-time and performing rule-based packet analysis [12, 13]. These rules act as signatures for various types of attacks. Each packet passing through SNORT is thoroughly analyzed to identify matches with predefined rules. In their work [14], M. Almseidin et al. propose a novel approach for detecting slow port scanning using a Fuzzy Rule Interpolation (FRI) rule set, which also determines the maliciousness level of detected attacks [15]. These rules are based on the following parameters:

- Number of Sent Packets (NSP) between the source and destination.
- Average Time between Packets (ATP) received by the victim, in milliseconds.
- Number of Packets Received (NPR) by the destination victim per second.

The majority of approaches proposed in the literature for detecting slow port scanning are focused on identifying slow port-scanning attacks within a static period. Mehr u Nisa et al. [16] proposed a technique to detect slow port-scanning attacks not only during static time intervals but also all attacks conducted with a gradual increase or decrease in duration over time. The proposed system is divided into four main modules. In the first module, real-time data packets are captured from the live network for analysis. In the second module, the captured data is analyzed to detect signs of port scanning and labeled accordingly. The third module categorizes the labeled packets into parallel and single scans based on the scanner's IP address and other selected features. Finally, in the last module, a decision is made based on time duration analysis to determine whether the scan was a fast or slow attack. The generated reports can then be used to block the attacker's IP address or take other necessary measures [17].

E.S. Sagatov et al., in their work [18] presented methods for detecting and countering the initial stages of cyber-attacks, including TCP and UDP port scanning. The proposed methods analyze outgoing traffic to identify response packets such as ICMP 3.3 and TCP RST, which indicate the onset of an attack. The authors also described two countermeasures based on developed modules for software-defined network controllers and Linux OS utilities. Testing of the developed methods was conducted on a cybersecurity testbed and demonstrated that the accuracy of detecting open TCP ports did not exceed 15%, while for other ports (closed TCP ports and UDP ports of any type), the accuracy remained below 2% [19, 20].

E. K. Baah [21] employed seven machine learning classifiers for port scan detection after successfully applying the Principal Component Analysis (PCA) algorithm for dimensionality reduction and selecting the most relevant features. A comparison of the results from various models and prior studies identified the XGBoost model as the best classifier, achieving the highest accuracy of 99.98%, with no false positives detected, a precision of 99.99%, a recall of 99.98%, and an Area under the Curve (AUC) of 99.99% [22, 23].

M. Ring et al. [24] propose an innovative approach to preprocessing streaming data, designed to detect slow port scanning. The preprocessing process generates new objects based on domain knowledge and network structure collected over a specific period. The computed objects are used as input data for further analysis, and based on these, two distinct approaches for detecting slow port scans have been proposed. The first approach employs sequential hypothesis testing, while the second utilizes classification algorithms [25]. The proposed methods were tested on the CIDDS-001 dataset.

Conclusion

The work performed is devoted to studying the efficiency of statistical methods for processing time series Z-score and modified Z-score in detecting PortScan and DDoS attacks. The results show that the modified Z-score is more relevant for detecting anomalies in time series with asymmetric distributions. Experiments have shown that DDoS attacks can be easily detected using statistical methods. With port

scanning detection, the performance of statistical methods depends on the type of scanning. Analysis of the results shows high performance in detecting fast scans, but complete inefficiency for slow scanning. Finding out the reasons for this behavior shows a local violation of the stationary state of time series in the time interval of fast scanning, which is reflected in the high KPSS coefficient. From all of the above, two conclusions follow – slow scanning does not differ from ordinary safe traffic in terms of statistical characteristics of time series. Analysis of local stationarity in time series can be taken as a basis for detecting fast port scans.

References

1. Scaranti G.F., Carvalho L.F., Barbon S., Lloret J., Proença M.L. Unsupervised online anomaly detection in Software Defined Network environments. *Expert Systems with Applications*. 2022;191:116225. https://doi.org/10.1016/j.eswa.2021.116225
2. Rookard C., Khojandi A. Unsupervised Machine Learning for Cybersecurity Anomaly Detection in Traditional and Software-Defined Networking Environments. *IEEE Transactions on Network and Service Management*. 2025;22(2):1129-1144. https://doi.org/10.1109/TNSM.2024.3490181
3. Birkinshaw C., Rouka E., Vassilakis V.G. Implementing an intrusion detection and prevention system using software-defined networking: Defending against port-scanning and denial-of-service attacks. *Journal of Network and Computer Applications*. 2019;136:71-85. https://doi.org/10.1016/j.jnca.2019.03.005
4. Ayodele B., Buttigieg V. SDN as a defence mechanism: a comprehensive survey. *International Journal of Information Security*. 2024;23(1):141-185. https://doi.org/10.1007/s10207-023-00764-1
5. Abrantes R., Mestre P., Cunha A. Exploring Dataset Manipulation via Machine Learning for Botnet Traffic. *Procedia Computer Science*. 2022;196:133-141. https://doi.org/10.1016/j.procs.2021.11.082
6. Hanzlik L., Kutyłowski M., Yung M. Hard Invalidation of Electronic Signatures. In: Lopez J., Wu Y. (eds.) Information Security Practice and Experience. ISPEC 2015. *Lecture Notes in Computer Science*. Vol. 9065. Cham: Springer; 2015. p. 421-436. https://doi.org/10.1007/978-3-319-17533-1_29
7. Ono D., Guillen L., Izumi S., Abe T., Suganuma T. A proposal of port scan detection method based on Packet-In Messages in OpenFlow networks and its evaluation. *International Journal of Network Management*. 2021;31(6):e2174. https://doi.org/10.1002/nem.2174
8. Bou-Harb E., Debbabi M., Assi C. Cyber scanning: a comprehensive survey. *IEEE Communications Surveys & Tutorials*. 2014;16(3):1496-1519. https://doi.org/10.1109/SURV.2013.102913.00020
9. Hartpence B., Kwasinski A. Combating TCP Port Scan Attacks Using Sequential Neural Networks. In: 2020 International Conference on Computing, Networking and Communications (ICNC). Big Island, HI, USA: IEEE Press; 2020. p. 256-260. https://doi.org/10.1109/ICNC47757.2020.9049730
10. Hirsi A., et al. Comprehensive Analysis of DDoS Anomaly Detection in Software-Defined Networks. *IEEE Access*. 2025;13:23013-23071. https://doi.org/10.1109/ACCESS.2025.3535943
11. Al-Haija Q.A., Saleh E., Alnabhan M. Detecting Port Scan Attacks Using Logistic Regression. In: 2021 4th International Symposium on Advanced Electrical and Communication Technologies (ISAECT). Alkhobar, Saudi Arabia: IEEE Press; 2021. p. 1-5. https://doi.org/10.1109/ISAECT53699.2021.9668562
12. Sarkar S., Roychowdhury S., Das A., Singh B.K. A Hybrid Intrusion Detection Framework for Hypervisor-Based MitM Attack Detection in Medical Cyber-Physical Systems: Leveraging PCA, Anomaly Detection, and KNN. In: 2025 International Conference on Artificial Intelligence and Machine Vision (AIMV). Gandhinagar, India: IEEE Press; 2025. p. 1-6. https://doi.org/10.1109/AIMV66517.2025.11203452
13. Nguyen V.Q., Ngo L.T., Nguyen V.H., Nguyen L.M., Le-Khac N. -A. A Deep Metric Learning Approach for Cyber Reconnaissance Detection. In: 2024 1st International Conference On Cryptography And Information Security (VCRIS). Hanoi, Vietnam: IEEE Press; 2024. p. 1-7. https://doi.org/10.1109/VCRIS63677.2024.10813453
14. Almseidin M., Al-Kasassbeh M., Kovacs S. Detecting Slow Port Scan Using Fuzzy Rule Interpolation. In: 2019 2nd International Conference on new Trends in Computing Sciences (ICTCS). Amman, Jordan: IEEE Press; 2019. p. 1-6. https://doi.org/10.1109/ICTCS.2019.8923028
15. Sangeen M., Bhatti N.A., Kifayat K. PortScout: A Communication Flow-Based Approach to Detect Port Scanning Evasion Attacks. In: ICC 2025 – IEEE International Conference on Communications. Montreal, QC, Canada: IEEE Press; 2025. p. 3045-3050. https://doi.org/10.1109/ICC52391.2025.11160775
16. Nisa M., Kifayat K. Detection of Slow Port Scanning Attacks. In: 2020 International Conference on Cyber Warfare and Security (ICCWS). Islamabad, Pakistan: IEEE Press; 2020. p. 1-7. https://doi.org/10.1109/ICCWS48432.2020.9292389
17. Nithya N., Sakthimuneeswaran S. Combined Approach using Multiattention Graph Convolution Network towards prevention, detection and classification of Phishing attack. In: 2025 2nd International Conference on New Frontiers in Communication, Automation, Management and Security (ICCAMS). Bangalore, India: IEEE Press; 2025. p. 1-6. https://doi.org/10.1109/ICCAMS65118.2025.11233973

18. Sagatov E.S., Mayhoub S., Sukhov A.M., Esposito F., Calyam P. Proactive Detection for Countermeasures on Port Scanning based Attacks. In: 2021 17th International Conference on Network and Service Management (CNSM). Izmir, Turkey: IEEE Press; 2021. p. 402-406. https://doi.org/10.23919/CNSM52442.2021.9615577
19. Seshapriyan T., Dinesh S.M., Ponsam J.G. SentinelScan: Advanced Network Scanner and Packet Detection Suite. In: 2024 8th International Conference on Inventive Systems and Control (ICISC). Coimbatore, India: IEEE Press; 2024. p. 253-258. https://doi.org/10.1109/ICISC62624.2024.00050
20. Huang J,. et al. Research on detection techniques for scanning attacks in software-defined network environments. In: 2023 4th International Conference on Computer Engineering and Application (ICCEA). Hangzhou, China: IEEE Press; 2023. p. 115-118. https://doi.org/10.1109/ICCEA58433.2023.10135250
21. Baah E.K., et al. Enhancing Port Scans Attack Detection Using Principal Component Analysis and Machine Learning Algorithms. In: Ahene E., Li F. (eds.) Frontiers in Cyber Security. FCS 2022. *Communications in Computer and Information Science*. Vol. 1726. Singapore: Springer; 2022. p. 119-133. https://doi.org/10.1007/978-981-19-8445-7_8
22. Saranya T., et al. Performance Analysis of Machine Learning Algorithms in Intrusion Detection System: A Review. *Procedia Computer Science*. 2020;171:1251-1260. https://doi.org/10.1016/j.procs.2020.04.133
23. Camacho J., et al. Group-Wise Principal Component Analysis for Exploratory Intrusion Detection. *IEEE Access*. 2019;7:113081-113093. https://doi.org/10.1109/ACCESS.2019.2935154
24. Ring M., Landes D., Hotho A. Detection of slow port scans in flow-based network traffic. *PLOS ONE*. 2018;13(9): e0204507. https://doi.org/10.1371/journal.pone.0204507
25. Ring M., Wunderlich S., Grüdl D., Landes D., Hotho A. A Toolset for Intrusion and Insider Threat Detection. In: Palomares Carrascosa I., Kalutarage H., Huang Y. (eds.) Data Analytics and Decision Support for Cybersecurity. Data Analytics. Cham: Springer; 2017. p. 3-31. https://doi.org/10.1007/978-3-319-59439-2_1

Submitted 12.04.2025; approved after reviewing 26.04.2025; accepted for publication 11.07.2025.

Поступила 12.04.2025; одобрена после рецензирования 26.05.2025; принята к публикации 11.07.2025.

About the author:

Adeyemi Marc Aurele Emmanuel Djeguede, Postgraduate Student of the Department of Mathematical Modeling and Artificial Intelligence, Faculty of Science, Peoples' Friendship University of Russia named after Patrice Lumumba (6 Miklukho-Maklaya St., Moscow 117198, Russian Federation), **ORCID: https://orcid.org/0000-0002-8476-8994,** djeguede.marc@gmail.com

The author has read and approved the final manuscript.

Об авторе:

Джегюеде Марк Аурele, аспирант кафедры математического моделирования и искусственного интеллекта факультета физико-математических и естественных наук, ФГАОУ ВО «Российский университет дружбы народов имени Патриса Лумумбы» (117198, Российская Федерация, г. Москва, ул. Миклухо-Маклая, д. 6), **ORCID: https://orcid.org/0000-0002-8476-8994,** djeguede.marc@gmail.com

Автор прочитал и одобрил окончательный вариант рукописи.