JURNAL SYNTAX TRANSFORMATION
Printed ISSN: 2721-3854 | Electronic ISSN: 2721-2769
Volume 5, No. 9 September, 2025
p-ISSN 2721-3854 | e-ISSN 2721-2769

Anomaly Detection in VPN (Virtual Private Network) Access Using Machine Learning Algorithms

Budi Prasetya, Hari Soetanto, Dimas Prayogo Mahardhika
Universitas Budi Luhur, Indonesia
Email: 2311600916@student.budiluhur.ac.id, hari.Soetanto@budiluhur.ac.id, 2331600565@student.budiluhur.ac.id

ABSTRACT
Network security is crucial in information technology systems, particularly in the banking sector, which handles sensitive data. This study successfully designed and implemented a VPN access anomaly detection prototype using log data from a Fortianalyzer device at Bank JTrust Indonesia. The developed method utilizes the DBSCAN algorithm with parameters optimized through Differential Evolution (DE) to improve clustering quality and anomaly detection accuracy. The research process includes data preprocessing, normalization, parameter optimization, and result visualization. Evaluation shows that this prototype can detect 10 anomaly points with striking characteristics, such as extreme access duration, unusual frequency, and imbalance in the volume of data sent and received. The detection accuracy rate reached 95.88% with a low false positive rate, indicating the effectiveness of this method in separating normal and anomalous access patterns. These findings confirm that the combination of data normalization, DE parameter optimization, and the DBSCAN clustering algorithm is highly suitable and reliable for responsive detection of cyber threats and supports the development of automated mitigation system integration in the future. Thus, this method is expected to strengthen cybersecurity in the banking environment through efficient and adaptive detection of suspicious VPNs.
Keywords: Network security, Anomalous access, VPN, Anomaly detection

INTRODUCTION

Network security is a crucial aspect of information technology management, especially in the embedded sector that handles sensitive and confidential data. Increasingly complex and sophisticated cyber threats demand adaptive and proactive security strategies. Jtrust Bank, as a leading financial institution, relies on Virtual Private Network (VPN) technology to ensure the security of remote connections and protect data from external threats (Alwhbi et al., 2024).

While VPNs provide data encryption and secure connection paths, these technologies are not completely immune to attacks (Chen & Li, 2011). One of the threats that often occurs is the exploitation of legitimate VPN accounts that exhibit anomalous activity patterns (Dong et al., 2019). This kind of activity, if not detected early, can lead to significant risks such as data leaks, sabotage, or identity theft (Chahal & Kaur, 2016; Diphoko, 2024; Foster et al., 2018; Karczmarek et al., 2020). Conventional detection methods often fail to recognize the anomalous patterns that grow increasingly complicated as cyberattacks evolve (Aslan et al., 2023).

According to a report by the Financial Services Authority (OJK), cyberattacks in the Indonesian banking sector caused financial losses of IDR 246.5 billion from the first semester of 2020 to the first semester of 2021 (Andani, 2023). Globally, the International Monetary Fund (IMF) estimates that losses due to cybercrime in the financial services sector reach around USD 100 billion per year. This data confirms the importance of improving cybersecurity in the banking sector to protect financial assets and institutional reputations (Liputan6.com, 2021).

The machine learning approach has become a promising solution for detecting patterns that are difficult to identify manually. Algorithms such as Density-Based Spatial Clustering of Applications with Noise (DBSCAN) have been used to recognize normal access patterns and

detect deviant activity. Additionally, the k-means algorithm offers an efficient approach to grouping access patterns based on specific features, allowing for more granular anomaly detection (Dong et al., 2019).

The k-means algorithm groups data into a number of clusters (k) based on the proximity of feature values (Sikos, 2020; Setiawan, 2020; Yuan et al., 2024; Zamanian et al., 2017). This algorithm is popular because of its simplicity and computational speed. In the context of anomaly detection, k-means can help identify network activity that deviates from the pattern of the main clusters. Research by Laskar et al. (2021) shows that the combination of k-means and isolation forest is effective in detecting anomalies in big data industry scenarios (Laskar et al., 2021).

In addition, the DBSCAN algorithm has the advantage of detecting outliers or data that lies outside the main group (noise). This algorithm works based on the density of the surrounding data, making it suitable for detecting anomalies hidden among VPN access patterns. Research by Botts (2020) found that variants of the DBSCAN algorithm are effective in detecting anomalous behavior in ship movement data (Botts, 2021).

Previous research has also demonstrated the effectiveness of the machine learning approach for detecting network anomalies. A study by Andani (2023) applied machine learning algorithms in detecting network anomalies in the computer network environment of XYZ University, showing an increase in detection accuracy (Andani, 2023). However, this approach still faces challenges in terms of high false positive rates. Therefore, a combination of algorithms such as k-means and DBSCAN has the potential to improve the accuracy and efficiency of detection.

The purpose of this study is to develop and evaluate a machine learning–based anomaly detection model for VPN access at Jtrust Bank using k-means and DBSCAN algorithms. The benefits of this research are twofold: theoretically, it contributes to the growing body of knowledge in cybersecurity and machine learning applications, particularly in the financial sector where VPN usage is critical; practically, it provides Jtrust Bank with a more adaptive and intelligent security framework that enhances data protection, mitigates risks of cyberattacks such as data leaks or identity theft, and supports more efficient network management.

RESEARCH METHOD

This research was designed to achieve the objectives of the study on anomaly detection in VPN access logs using machine learning algorithms. The approach consisted of systematic steps to ensure that the selected model functioned effectively and efficiently in detecting anomalies. The study combined data analysis experiments with model optimization of machine learning algorithms.

Sample selection was carried out using the purposive sampling method, as the research focused on VPN access log data with specific characteristics relevant to evaluating anomaly detection models. The population comprised all VPN access log data generated by Bank Jtrust's network system over a defined period. The instruments used included hardware, software, and datasets. To evaluate model performance, one algorithm was selected and tested.

Research Steps

The steps of this study were systematically designed to achieve the research objective, namely the selection of a machine learning-based anomaly detection model for Jtrust Bank VPN access logs. Here are the stages of the research:

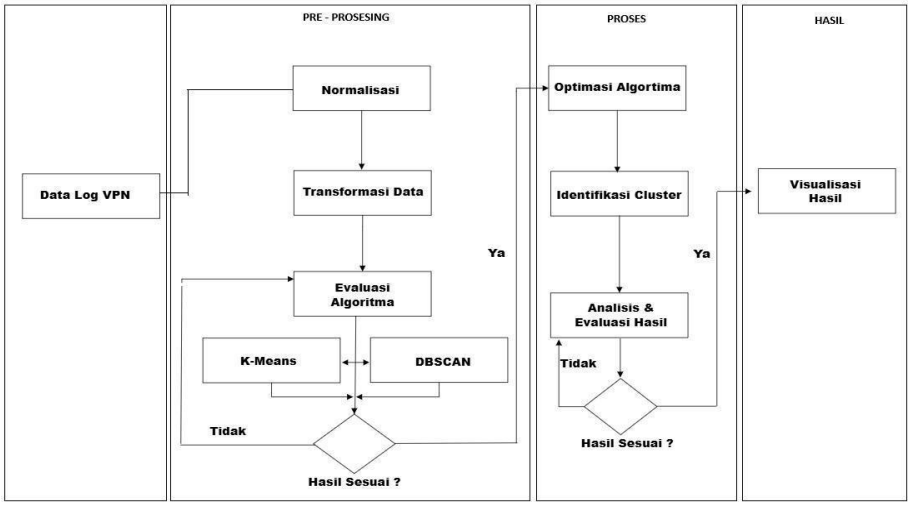


Figure 1. Research Steps

Data Log VPN

This stage is the starting point of the research, where VPN log data is collected from the system at Bank Jtrust. This data includes important information such as IP Address, Access time, User ID, Usage Pattern, Access frequency, Access location, Session duration, Number of data used.

Pre Processing

1) Normalization is the data that has been collected in normalization to ensure that all features are at the same scale.
2) Data Transformation i.e. data is transformed into a suitable format for further analysis.
3) Algorithm evaluation is at this stage, the Machine Learning algorithm is evaluated to determine the most suitable algorithm for anomaly detection. Based on the type of log data, namely unlabeled log data and according to the purpose of making the model, namely detecting the level of data density and detecting the level to accuracy of anomalies. Furthermore, in evaluation, what type of algorithm matches the data and the purpose of creating this model.
4) K-Means is a clustering algorithm used to divide a set of data into K clusters based on the proximity of the data to the center of the cluster (centroid).
5) DBSCAN is a clustering algorithm that groups data points based on density. This algorithm can find clusters with arbitrator shapes and can also identify data points that are considered to be noise (outliers).
6) The result is appropriate if the results of the algorithm evaluation are in accordance with the data criteria and the purpose of the model that have been set, it is used for the next stage. Otherwise, the evaluation process is repeated until a suitable algorithm is found

Process

1) Algorithm optimization is the selected algorithm that is optimized to improve the accuracy and efficiency of anomaly detection.
2) Cluster Identification is data that is grouped using an algorithm that has been optimized to identify clusters that represent normal access patterns and are outside normal or anomalous access.
3) Analysis and evaluation of results are the results of clustering in analysis and evaluation to ensure that the clusters formed are in accordance with expectations and data outside the cluster is identified as anomalous data.

4) The results are appropriate, that is, if the results of the analysis and evaluation are appropriate, then the process continues to the visualization stage of the results. Otherwise, the cluster identification process is repeated until the corresponding result is achieved.

Result

1) The visualization of the results of the clustering and anomaly detection process is visualized to provide a clear and easy-to-understand picture of the model created.

RESULTS AND DISCUSSION

This research aims to provide a practical and interactive solution in detecting anomalies in Virtual Private Network (VPN) network access data. The machine learning method chosen in this propotype is DBSCAN (Density Based Spatial Clustering of Applications with Noise), a density-based clustering algorithm that is widely recognized as able to detect outliers or anomalies effectively, without the need to pre-define the expected number of clusters.

Hardware Specifications

Table 1. Hardware Specifications

Component	Specifications
Processor	Processor AMD FX-8320
Ram	12 GB
OS	Windows 11 Pro

Software Environment

Table 2. Software Environment

Component	Specifications	Support
Python	Version 3.13	Linux, Windows 10,11 64,32 bit, macOS
Visual Studio Code	Version 1.101.2	Windows 10.11 64.32 bit, macOS
Streamlit	Version 3.13	Windows 10,11 64,32 bit, Linux, macOS

Prototype Process Flow

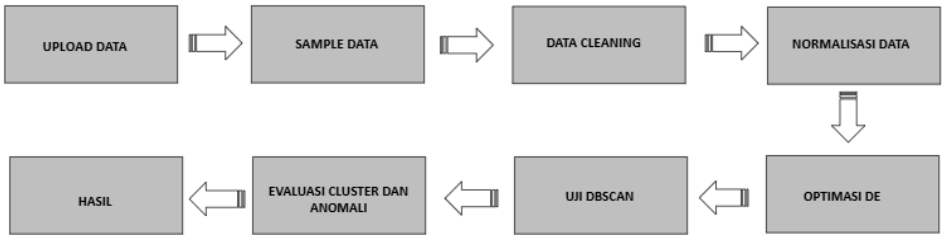


Figure 2. Prototype Process Flow

Figure 2 describes the flow of the prototype process where this anomaly detection system aims to analyze 1555 rows of VPN access log data from Bank Jtrust Indonesia (period 2024-2025, sourced from Fortianalizer) to identify patterns of abuse or suspicious activity. The process begins with uploading, cleaning, and normalizing the log data. At the heart of this process is parameter optimization (eps and minpts) using the Differential Evolution (DE) algorithm, which is then used to run clustering with DBSCAN. DBSCAN functions to group data, where points that do not belong to the cluster (noise) are identified as anomalies. All clustering and anomaly detection results are evaluated to ensure their quality, with the final result being the number of clusters and anomaly data detected.

This VPN log dataset contains 10 attribute columns, which represent various aspects of the VPN access session

Table 3. VPN log dataset

Number	Column
1	Date
2	Time
3	User_id
4	Access_frequency_perhari
5	Duration_seconds
6	Client_ip
7	Server_ip
8	Bytes_sent
9	Bytes_received
10	Location

Upload Data

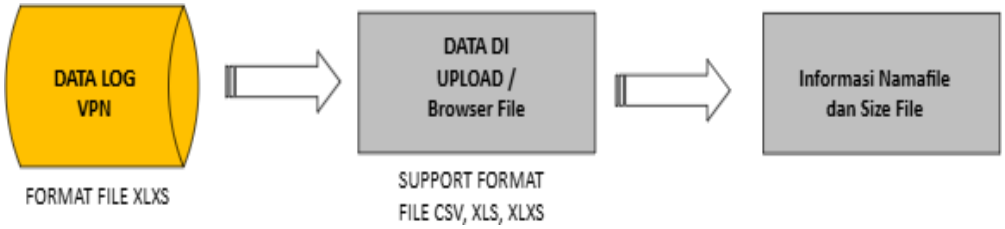


Figure 3. Data Upload Topology

The explanation of figure 3 topology of uploading data in this process is, vpn access log data in excel file format is uploaded to the dashboard browser view, for formats available in csv, xls, and xlxs that can be managed. Then after the data is uploaded, the results of file name information and file size information are displayed.

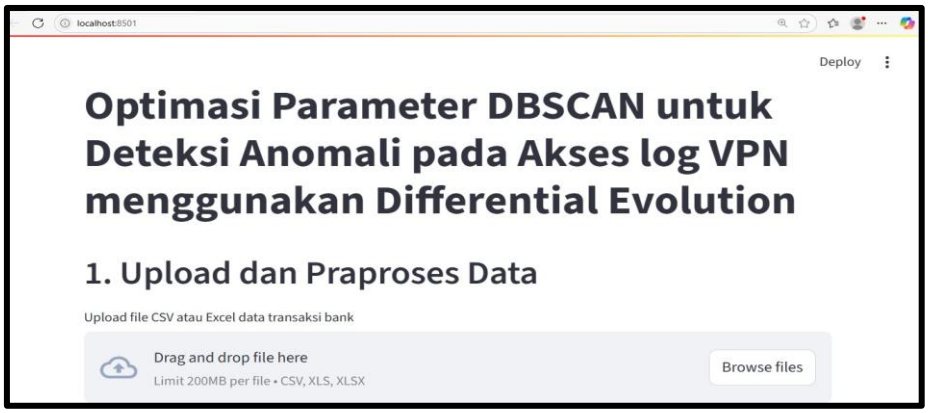


Figure 4. Upload Data

In Figure 4, the Data Upload Stage is implemented on the Streamlit dashboard to facilitate users to easily enter VPN log datasets. Users can choose the Drag and Drop method or use the "Browser Files" button. The system supports CSV, XLS, and XLXS formats with a maximum file size limit of 200MB. Once the file is successfully uploaded, the application uses Pandas libraries (such as pd.read_csv or pd.read_excel) to read the data and automatically prepares it for the next initial preprocessing stage, which includes missing value handling, sampling, normalization, and transformation (Mboweni et al., 2023; Schummer et al., 2024; Shim, 2021; Shorewala, 2021).

Sample Data

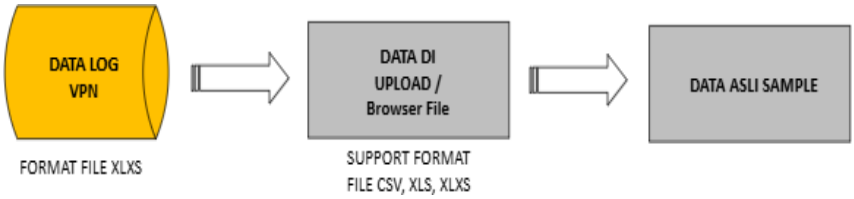
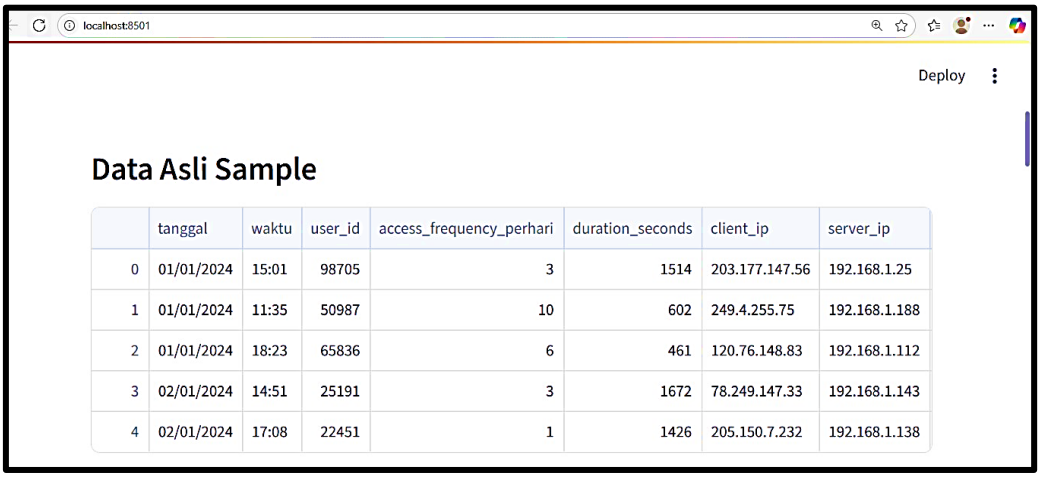


Figure 5. Sample Native Data Process Topology

In Figure 5 Sample Original Data Process Topology, the sample data visualization process begins with VPN log data which is generally in the form of XLSX files. These files are uploaded through the app's user interface, where users can select and upload files in supported formats (CSV, XLS, or XLSX). Once the file is successfully uploaded, the system will read the original data. At this stage, the data is extracted and immediately processed into a sample of data that is ready to be visualized and analyzed. The success of this process ensures that the original data from the VPN access log is successfully displayed in the appropriate format and ready for the next stage, namely data cleaning, normalization, and anomaly analysis.



	tanggal	waktu	user_id	access_frequency_perhari	duration_seconds	client_ip	server_ip
0	01/01/2024	15:01	98705	3	1514	203.177.147.56	192.168.1.25
1	01/01/2024	11:35	50987	10	602	249.4.255.75	192.168.1.188
2	01/01/2024	18:23	65836	6	461	120.76.148.83	192.168.1.112
3	02/01/2024	14:51	25191	3	1672	78.249.147.33	192.168.1.143
4	02/01/2024	17:08	22451	1	1426	205.150.7.232	192.168.1.138

Figure 6. Sample Data

In figure 6 sample data explains that after the data upload process is successfully carried out, the system displays a visual display in the form of a sample data table on the application interface page. This table explicitly shows some of the initial entries from the original dataset, which include various key columns that reflect user activity on VPN logs.

Data Cleaning

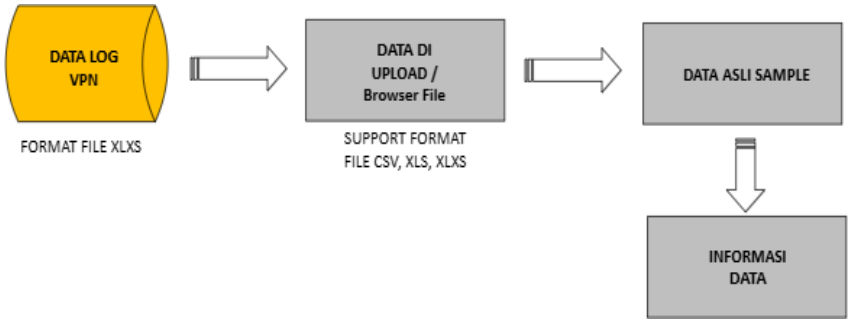


Figure 7. Data Cleaning Process

In figure 7 Data Cleaning Process shows, the data cleaning process begins after the original sample data is successfully uploaded and displayed. The data coming from VPN logs in XLSX format is processed to extract relevant information and clean the data from missing values, inconsistencies, and formatting errors. This process includes data quality checking, deletion of corrupted and invalid data. So the result of the cleaning process in figure 7 of the data cleaning is a collection of clean data that is ready to be used in the normalization process, to produce an accurate and reliable analysis.

Data Normalization

Data normalization is an important stage in data pre-processing that aims to change the scale of numerical variables to be in a common or uniform range. This process ensures that all features have a balanced contribution to the machine learning algorithm, specifically the DBSCAN clustering algorithm. In this anomaly detection prototype, normalization plays an important role in improving the accuracy and stability of anomaly detection in the VPN access logs dataset.

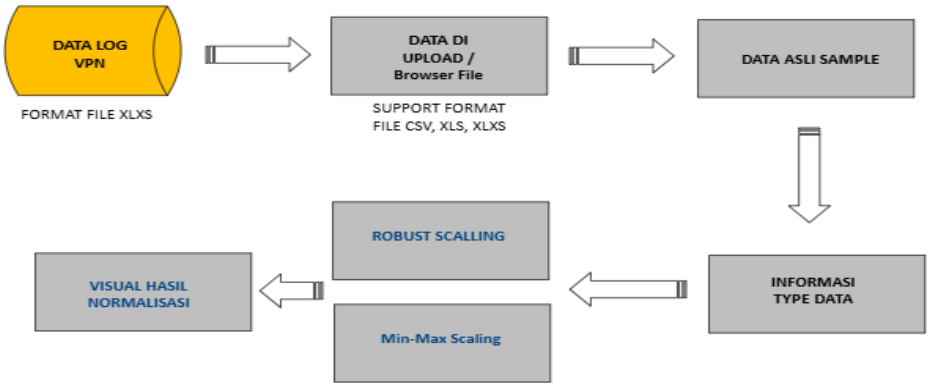
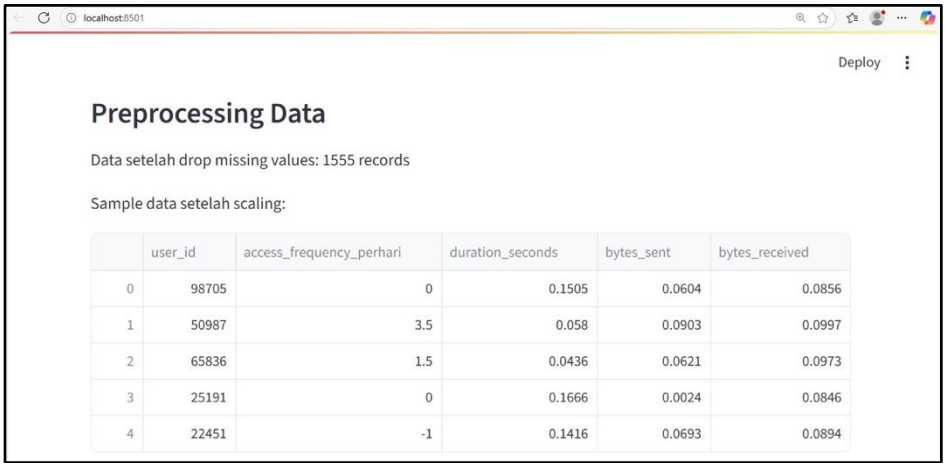


Figure 8. Normalization Process

In Figure 8 the data normalization process begins after the VPN log data is successfully uploaded in a supported format (CSV, XLS, or XLSX) and generates the original sample data. The next crucial stage is to identify the data type of each column to determine the most appropriate normalization method. Based on the analysis of the characteristics of the data, two main methods were applied: Robust Scaling was used specifically for access_frequency_daily columns because these columns had the potential to contain outliers, where Robust Scaling was more effective because it used the median and interquartile ranges. Meanwhile, Min-Max Scaling is applied to the duration_seconds, bytes_sent, and bytes_received columns. This method was chosen for relatively homogeneous quantitative data and aimed to transform the feature value to the range [0,1] proportionally. This whole process ensures the scale of different variables becomes uniform, which is an important prerequisite for anomaly analysis with DBSCAN.



	user_id	access_frequency_perhari	duration_seconds	bytes_sent	bytes_received
0	98705	0	0.1505	0.0604	0.0856
1	50987	3.5	0.058	0.0903	0.0997
2	65836	1.5	0.0436	0.0621	0.0973
3	25191	0	0.1666	0.0024	0.0846
4	22451	-1	0.1416	0.0693	0.0894

Figure 9. Normalization Data visualization

In Figure 9 Normalization Data Visualization, the access_frequency_daily column represents the frequency of access per day that tends to have variable values and potentially contains outliers (very high or very low extreme values). Robust scaling uses the median and interquartile range (iqr) so that it is more resistant to outliers than other methods. For the duration_ second column, bytes_second and bytes_received, this column represents quantitative values that are continuously scaled and relatively homogeneous without extreme outliers that interfere with the distribution of data. Min max scalling changes this data cell to the range 0 and 1 so that it makes it easier to represent numerically for the dbscan algorithm so that the features of this feature are proportional and measurable, the min max scaling is chosen because it adjusts the data type which if using other methods the results are less intuitive or the distribution of features becomes less proportional in the context of a definite range. After normalization, the data in these columns have a more standardized value and are ready for the next analysis process. That is the optimization process using the differential evolution method.

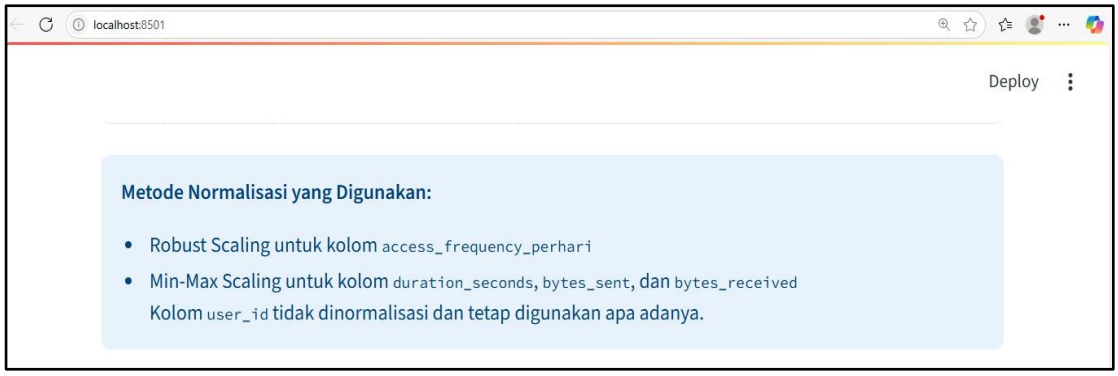


Figure 10. Data Normalization Methods

Differential evolution parameter optimization

Figure 11. Differential Evolution Parameter Optimization Process

The anomaly detection process relies heavily on a density-based clustering algorithm, DBSCAN, which is sensitive to its two main parameters: eps (epsilon, maximum distance of the environment) and min_samples (the minimum number of points to form a cluster). Errors in determining the values of these two parameters can result in meaningless cluster formation or categorize too much data as anomalies.

Therefore, the Differential Evolution (DE) algorithm is used to optimize DBSCAN parameters. DE is a population-based optimization algorithm that was chosen because it has reliable global search capabilities, allowing it to efficiently explore the parameter space and avoid getting stuck on local optimization. In this prototype, the DE is tasked with finding the best combination of values (eps, min_pts) that maximize clustering performance and anomaly detection. This performance is measured through objective functions designed to maximize clustering internal evaluation metrics, such as the Silhouette Score and the Davies-Bouldin Index, so that DBSCAN can identify clusters and detect anomalies with optimal accuracy.

Figure 12. Parameter Optimization

Figure 12 Parameter Optimization shows that each iteration shows that the adaptive DE algorithm searches for variable parameter values to try to get the best DBSCAN configuration. These values can then be evaluated based on clustering and anomaly metrics. Number of clusters formed, and number of anomalies found

Test DBSCAN parameters

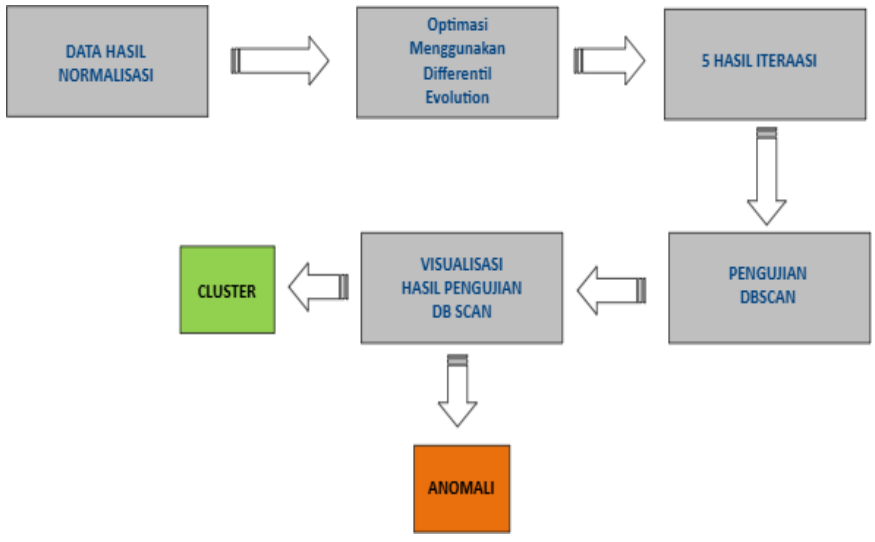


Figure 13. DBSCAN Test Process

In Figure 13 of the DBSCAN Test Process explains about the data input of optimization results using DE, the best process of the iteration results is selected to perform the DBSCAN test.

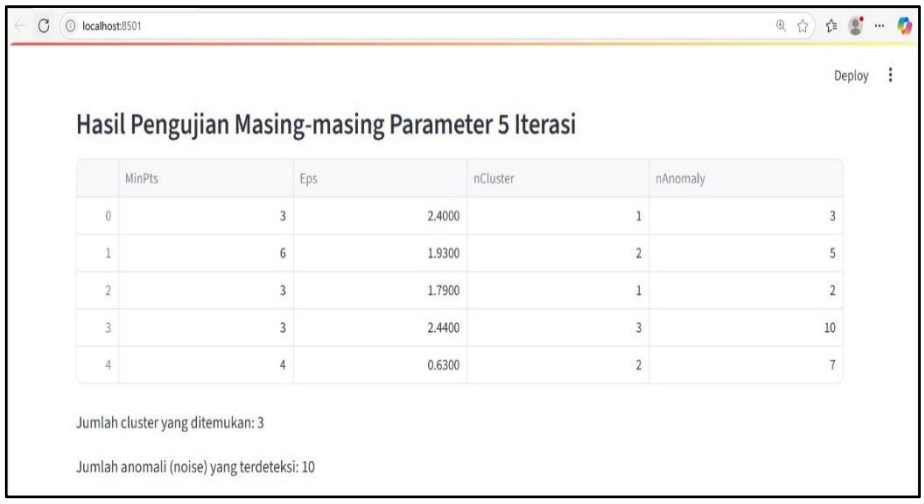


Figure 14. Test Results

In Figure 14 the test results explain the final results of the test day, namely:

1) Cluster: A group of data that has similar characteristics according to DBSCAN
2) Anomaly: Data that does not fit into any cluster or that is statistically different so that it is considered an anomaly

Cluster and Anomaly Evaluation

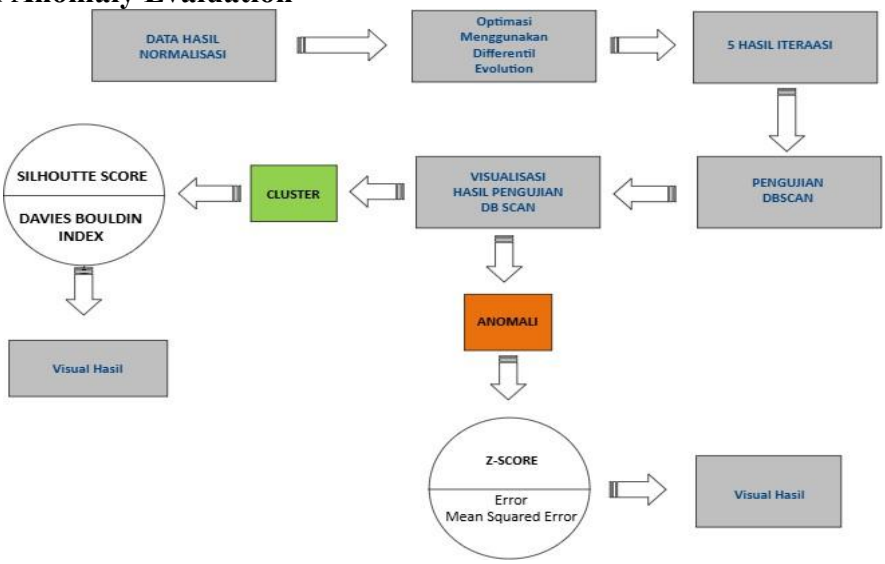


Figure 15. Cluster and Anomaly Evaluation Process

Evaluation of clustering results and anomaly detection is a critical stage in data analysis, especially when using the DBSCAN density-based clustering method. This evaluation aims to ensure that the results of clustering and anomalies truly represent real and meaningful patterns in the data, and that the anomalies detected are not just noise or modeling errors. In the context of prototypes, evaluation also provides scientific validation of the effectiveness of the methods used, as well as assisting in the interpretation of results by the user or tester.

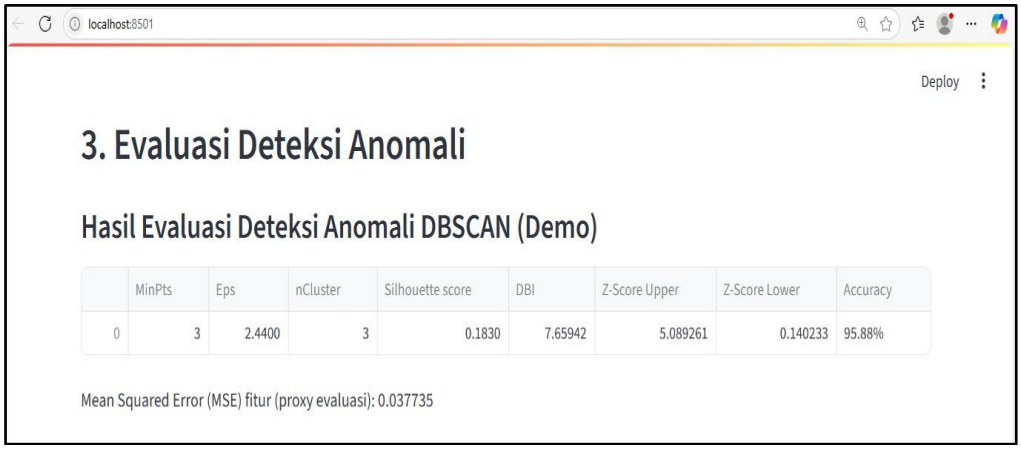


Figure 16. Evaluation of Anomaly Detection

Evaluated data

Evaluation was carried out on the output of DBSCAN which consisted of:

1) Cluster Label: each data point in the data is labeled integer as a cluster identity 0.1
2) Anomaly Label (outlier): a point that does not belong to any cluster is given a special label, -1 and is considered an anomaly
3) Data features: The data has gone through the preprocessing and normalization stages, including feature features such as access frequency per day, session duration, sent and received bytes used as the basis for cluster formation and anomaly identification

Cluster evaluation methods

The evaluation was carried out using metrics that did not require ground truth labels (original labels) and measured the quality of groupings statistically and geometrically.

Silhouette score

One of the evaluation methods used to assess the quality of clustering results, including groupings generated by algorithms such as DBSCAN. Silhouette score function Measures how well each data corresponds to the cluster it occupies compared to other clusters, Provides an overview of the cohesion and separation of the formed clusters, Silhouette score values range from -1 to 1, Where values close to 1 Data are very suitable and in accordance with the cluster itself, and are clearly separated from other clusters (good grouping). Values close to 0 Data are at the edge between the two clusters, indicating the cluster is not too different. Values close to -1 Data are most likely misplaced, better suited to other clusters than the cluster itself, signaling poor clustering. Silhouette score assesses how solid and discrete the resulting cluster is, allowing to determine the best parameters for optimal cluster and anomaly detection

Zscore (upper and lower)

Measures the distance of point data from the center of the cluster in standard units of deviation. Z-score is used to detect anomalies because data far from the center of the cluster has a high (positive) or low (negative) Z-score value. Upper and lower refer to the extreme limit (for example, Z>2 or Z<-2) to mark the data as an anomaly. A high Z-score indicates that the data is most likely an anomaly, while a value close to zero indicates that the data is within a normal cluster area.

Davies-Bouldin index

Measure the quality of clustering. The lower the DBI value, the better and denser the cluster. In much of the literature and data mining practitioners, a DBI value of < 1.5-2 is often

considered good clustering, and a value above 3 is generally considered suboptimal. By the way it works, calculate the distance ratio between clusters and the size of the cluster itself (such as mean distance and variance). The DBI estimates how separate the clusters are from each other; A high value indicates an overlapping cluster, while a low value indicates a good and clearly separate cluster.

Mean Squared error (MSE)

Measures how close the data is to the center of the cluster. The smaller the MSE, the better the cluster represents the data and the denser the cluster. By working to calculate the squared difference between the data and the centroid cluster, the average of all those squared differences gives the MSE value. A high MSE indicates widespread data from the cluster center and indicates poor clustering results. 0 to 2 low MSE — the model can reconstruct the data well and show high accuracy in anomaly detection and normal data. 3 to 5 MSEs are moderate—the model is somewhat less accurate; the data reconstruction begins to experience some deviations. More than 5 MSEs are less capable of accurately reconstructing data, showing many errors and possible difficulties in distinguishing anomalies from normal data.

Anomaly Evaluation

Evaluation of anomalies is a major part because the main purpose of the propoetype is to detect unusual activity

1) Number of anomaly points
   Checks the amount of data considered an anomaly (label -1) as a measure of DBSCAN sensitivity
2) Analysis of anomaly characteristics
   View the feature values of anomaly points to understand abnormal patterns, such as extremely high access frequency, unusual access duration, or volume abnormal bytes
3) Domain Validation
   Involves an assessment from a domain perspective, e.g. whether the anomaly pattern corresponds to signs of unnatural activity on VPN access (Extraordinary IP usage, bytes spikes and extreme duration)

Evaluation in the protype

Practically, this evaluation process includes

1) Retrieval of the output from DBSCAN
   Cluster labels and anomalies are obtained and stored
2) Cluster internal metric calculation
   Using the function function from the statistical data science library scikit-learn to calculate the value of silhouette score, Calinski and Davies Bouldin
3) Cluster size and feature statistics
   Perform static mean, median and feature distribution aggregation in each cluster
4) Anomaly identification and analysis
   Compile a list of anomaly points, analyze their feature features and compare them with normal clusters

Benefits of evaluation

Ensuring the quality of the analysis, by ensuring that the clustering results are not just random labels, but reflect real patterns, facilitate follow-up decisions with valid evaluation results that can be used for follow-up, conduct deeper anomaly investigations and improve the VPN network security system and in this evaluation become strong scientific evidence of the validity of the methods and the results obtained. And also Evaluation using Silhouette Score,

MSE, DBI, and Z-score provides benefits in assessing the quality and accuracy of clustering models and anomaly detection. The Silhouette Score helps ensure a cohesive and well-separated grouping of data, while the MSE measures the error rate in the data reconstruction, demonstrating the accuracy of anomaly detection. The DBI assesses how well clusters differ from each other, with low values indicating effective clustering. Z-score is useful in identifying outliers (anomalies) based on standard deviations from the data, thus increasing confidence in anomaly detection. This combination provides a comprehensive picture to ensure the model works optimally and reliably in accurately identifying anomalies. Below is an explanation of the evaluation:

Table 4. Explanation of the evaluation

Column	Explanation
MinPts	A DBSCAN parameter that indicates the minimum number of data points within the Eps radius to form a cluster. In this result, MinPts = 3, meaning that each cluster must have at least 3 data.
Eps	The maximum distance parameter (radius) that DBSCAN uses to determine the neighboring points in forming a cluster, here is worth 2.44
nCluster	The number of clusters formed after DBSCAN is run with these parameters. A value of 3 means that there are three main clusters found in the data.
Silhouette score	A clustering evaluation method that measures how well objects are grouped. A value of 0.1830 indicates clusters formed with light to moderate separation and cohesion. The silhouette score ranges from -1 to 1, numbers near 0 indicate clusters are not very clearly separated.
DBI (Davies-Bouldin Index)	A clustering evaluation index that measures the average similarity between clusters. A value of 7.65942 indicates that the cluster has a fairly high similarity between clusters, where a smaller value is better. This value can still be considered less than optimal.
Z-Score Upper	The upper limit of the Z-Score evaluation feature, which is 5.089261. Z-Score is a standard measure of deviation used to detect extreme data outliers/anomalies.
Z-Score Lower	The lower limit of the Z-Score evaluation feature (0.140233), being the lower threshold in anomaly detection.
Accuracy	The accuracy of anomaly detection based on the evaluation carried out, reaching 95.88%, indicates that the model is quite good at classifying anomalies and normal data.
Mean Squared Error (MSE)	It is recorded as a value of 0.037735, this is the average error value of the square used as a proxy to measure the accuracy of the model's predictions. A low value indicates the model is quite accurate in detecting anomalies

Interpretation of Evaluation Scores

Silhouette score and DBI are metrics that are often used to assess cluster quality. Although the silhouette score is relatively low (0.18), the accuracy of 95.88% shows that the model is still able to detect anomalies well. This may be because the anomaly data is very different from the primary cluster, so the anomaly accuracy is high even though the cluster is not very tight.

Algorithm Selection Results Analysis

To determine the best clustering algorithm to detect VPN access anomalies in the Jtrust Bank environment, a comprehensive evaluation process was carried out on two algorithms used to determine the selection of the best algorithm in the creation of this prototype, k-means and dbscan. This analysis is expected to provide a clear picture of the advantages and disadvantages of each algorithm based on technical and empirical performance aspects obtained from this propaction test:

Table 5. Algorithm selection analysis

Aspects	K-Means Algorithm	DBSCAN Algorithm
Objectives/Focus	Grouping data into a predetermined number of groups or (clusters) (K) based on the proximity of the feature value. Aims to identify network activity that deviates from the main pattern of the cluster. This algorithm is popular because of its simplicity and speed of computing. - The number of Clusters (K) determines the number of clusters to be formed. - Initialization method: a method to select an initial centroid such as k means that increases convergence. - Convergence criteria: conditions that determine when an algorithm stops iteration, usually based on minimal changes in centroid position	Groups data based on the density of surrounding data points, effectively identifying outliers or noise. Suitable for detecting anomalies hidden among normal VPN access patterns, especially on unstructured data. DBSCAN can find clusters of varying shapes and sizes without prior knowledge of the number of clusters.
Main parameters	Distance metric: Used to evaluate how well each data point is classified in a cluster	Epsilon (eps): The maximum distance between two points to be considered part of the same cluster. MinPts: The minimum number of dots in the epsilon radius to form a cluster. Optimization method: the optimization method on DBSCAN, optimized using Differential Evolution (DE) which aims to improve clustering quality and accuracy to detect anomalies with minimal accuracy 90%
Common normalization techniques applied	Min-Max Scaling: Converting a feature value into a specific range (e.g. 0–1), is important for data-scale-sensitive algorithms.	Robust Scaling: Uses the median and interquartile ranges, making it resistant to outliers. Applied to access_frequency_perhari on a prototype of this study. Min-Max Scaling: Converting the value of a feature to a specific range (i.e. 0-1) is applied to duration_seconds, bytes_sent and bytes_received in the propotype of this study.
Evaluation metrics	Elbow Method: A technique	Silhouette Score: Assesses the quality of

Aspects	K-Means Algorithm	DBSCAN Algorithm
	to determine the optimal number of clusters by plotting inertia against the number of clusters. Silhouette Score: Measures how well each data point fits its cluster compared to other clusters (value - 1 to 1, the higher the better). Davies-Bouldin Index (DBI): Measures the average similarity between clusters (the lower the better).	clustering, showing the cohesion and separation of clusters (values -1 to 1, the higher the better). Z-score: Measures how far a data point is from the mean of a cluster in standard deviation; A high Z-score can indicate an anomaly Mean Squared Error (MSE): Measures the average of the squared difference between the predicted value and the actual value Lower MSE indicates better cluster representation and higher anomaly detection accuracy. Davies-Bouldin Index (DBI): Measures the average similarity between clusters (the lower the better).
Performance on research proptypes (DBSCAN)	The research prototype focuses on the implementation and evaluation of DBSCAN, so performance results specific to K-Means are not provided in the context of this prototype. However, the study is aimed at evaluating both (DBSCAN and K-Means) to determine the most appropriate algorithm for VPN anomaly detection.	**Accuracy** Describes the proximity of the results obtained to the actual or expected value. The prototype successfully detected 10 anomaly points with different characteristics, such as extreme access duration, unusual frequency, and unbalanced data volume. The detection accuracy rate reached 95.88% with a low false positive rate. The value of the evaluation metric achieved for the DBSCAN model that was implemented was with the value Minpts = 3, Eps = 2.44, nCluster = 3, silhouette score = 0.1830, DBI = 7.65942, Z-score above = 5.089261, lower Z-score = 0.140233 and MSE = 0.037735

Anomaly Results

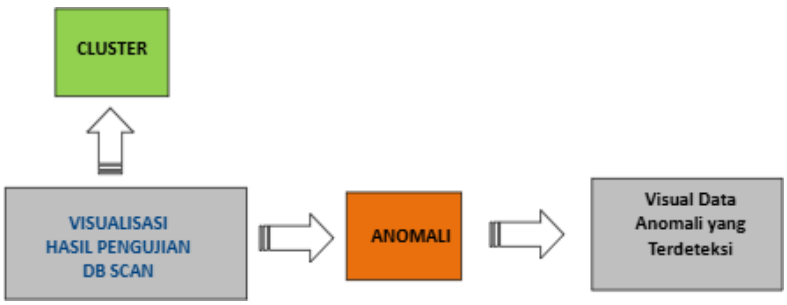
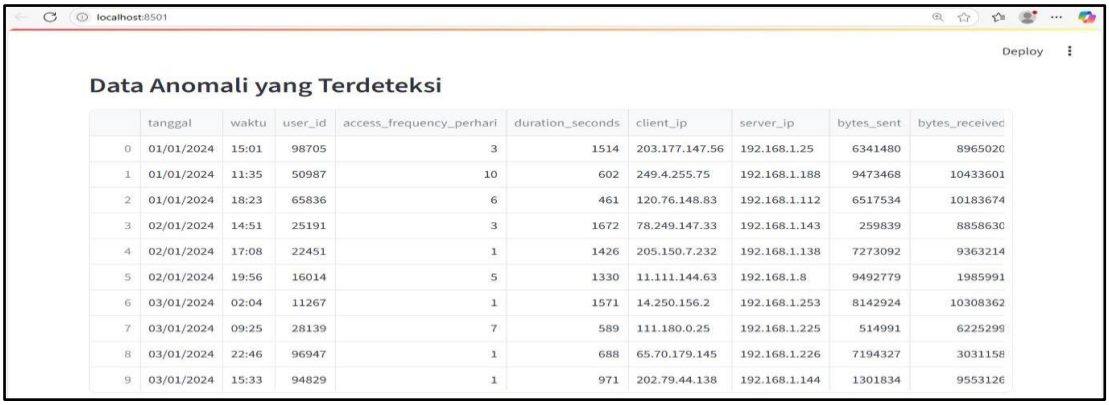


Figure 17. Visualization of Anomaly Results

In figure 17 Visualization of anomaly results explains that after the test process is carried out using DBSCAN on the normalized data and the parameter optimization using differential evolution, the next step is to identify anomalies in the data, namely:

1) Visualization of DBSCAN test results
   The test result data is marked by the results of the ncluster value anomaly
2) Identification of anomalies
   From the results of the visualization of anomalies, data that is not incorporated into the cluster is marked as noise by DBSCAN and identified as anomalies. This anomaly is data whose behavior is significantly different from clusters in general and is considered an outlier.
3) Anomaly output
   After the anomaly is identified, it can be seen in figure 17 the results of the anomaly detected. Anomaly data is visualized and presented, see Figure 18 anomaly data detected showing 10 data that are considered anomalies. So that it facilitates analysis and decision-making.

Figure 18. Anomaly Detected Results



	tanggal	waktu	user_id	access_frequency_perhari	duration_seconds	client_ip	server_ip	bytes_sent	bytes_received
0	01/01/2024	15:01	98705	3	1514	203.177.147.56	192.168.1.25	6341480	8965020
1	01/01/2024	11:35	50987	10	602	249.4.255.75	192.168.1.188	9473468	10433601
2	01/01/2024	18:23	65836	6	461	120.76.148.83	192.168.1.112	6517534	10183674
3	02/01/2024	14:51	25191	3	1672	78.249.147.33	192.168.1.143	259839	8858630
4	02/01/2024	17:08	22451	1	1426	205.150.7.232	192.168.1.138	7273092	9363214
5	02/01/2024	19:56	16014	5	1330	11.111.144.63	192.168.1.8	9492779	1985991
6	03/01/2024	02:04	11267	1	1571	14.250.156.2	192.168.1.253	8142924	10308362
7	03/01/2024	09:25	28139	7	589	111.180.0.25	192.168.1.225	514991	6225299
8	03/01/2024	22:46	96947	1	688	65.70.179.145	192.168.1.226	7194327	3031158
9	03/01/2024	15:33	94829	1	971	202.79.44.138	192.168.1.144	1301834	9553126

Figure 19. Anomaly Data Detected

The explanation in figure 19 of the anomalous data detected, shows the results of anomaly detection from VPN access log data using the DBSCAN method whose parameters have been optimized using differential evolution. The system managed to identify a total of 10 anomalies from the total data analyzed (Ayu & Anggraeni, 2012; Hasan, 2024). The number of anomalies detected shows that there are 10 data points that have significantly different characteristics than normal access patterns. These anomalies can be in the form of access sessions with very long and very short durations, unusual access frequencies, and unbalanced data volumes (Ibadirachman et al., 2024). These results are an early indicator that the method used is effective in separating normal and anomalous patterns in VPN access log data, so that it can be used as a basis for improving security systems and early detection of possible abuse and network attack patterns. Below I discuss the 10 anomaly findings and their reasons:

Table 6. VPN Access Log Data Anomaly Detection Results

Number	Key Characteristics	Reasons for the Anomaly
0	Frequency: 3/day, Duration: 1514 seconds, Bytes is huge, Location: Indonesia	Very long durations and large volumes of data do not conform to the majority pattern → are considered anomalies.
1	Frequency: 10/day, Duration: 602 seconds, Large bytes, Location: United States	High access frequency & relatively long duration → unusual behavior.
2	Frequency: 6/day, Duration: 461 seconds, Location: Japan	Different geographical locations (Japan) + moderately high access behavior → outliers.
3	Frequency: 3/day, Duration: 1672 seconds, Small sent bytes, large received	The duration of length & imbalance of data volume → different from other clusters.
4	Frequency: 1/day, Duration: 1426 seconds, Large Bytes, Location: Indonesia	Very low frequency + long duration → not in accordance with the general pattern.
5	Frequency: 5/day, Duration: 1330 seconds, Large bytes, IP: 11.111.144.63	Long duration + uncommon IP + large data volume → detected as an anomaly.
6	Frequency: 1/day, Duration: 1571 seconds, Location: Indonesia	The combination of very long duration & low frequency → far from the main cluster.
7	Frequency: 7/day, Duration: 589 seconds, Small bytes sent, large received, Location: Indonesia	The volume of data is unbalanced + the relative frequency is high → different from the majority.
8	Frequency: 1/day, Duration: 688 seconds, Location: Korea, Unbalanced bytes	Different geographic locations + abnormal data volumes → outliers.
9	Frequency: 1/day, Duration: 971 seconds, Location: Indonesia	Very low frequency + fairly long duration → only a few neighbors within a ε radius, so it is an anomaly.

CONCLUSION

This research successfully designed and implemented a prototype for detecting VPN access anomalies using log data from a Fortianalyzer device at Bank Jtrust Indonesia. By applying data pre-processing, normalization, and parameter optimization with Differential Evolution, the system optimized DBSCAN clustering parameters (MinPts and Eps) and achieved a detection accuracy of 95.88%. The model effectively identified 10 anomaly points with distinct characteristics, such as extreme access durations, unusual frequencies, and imbalanced data volumes, thereby separating normal from abnormal patterns. These findings demonstrate that the integration of DBSCAN and Differential Evolution is a reliable and effective method for anomaly detection, providing a strong foundation for enhancing VPN security through early detection of potential cyberattacks. However, further development is needed by incorporating deeper anomaly cause analysis and responsive mitigation systems to enable automatic preventive measures and strengthen network security management for data mining-based anomaly detection in the field of network security.

REFERENCES

Alwhbi, I. A., Zou, C. C., & Alharbi, R. N. (2024). Encrypted network traffic analysis and classification utilizing machine learning. Sensors, 24(11). https://doi.org/10.3390/s24113509

Andani, M. (2023). Implementation of machine learning algorithms in network anomaly detection: A case study on computer networks of XYZ University. Journal of Computer Science (JILKOM). https://mand-ycmm.org/index.php/jilkom/article/view/460

Aslan, Ö., Aktuğ, S. S., Ozkan-Okay, M., Yilmaz, A. A., & Akin, E. (2023). A comprehensive review of cyber security vulnerabilities, threats, attacks, and solutions. Electronics (Switzerland), 12(6). https://doi.org/10.3390/electronics12061333

Ayu, I., & Anggraeni, W. (2012). 144498-ID-implementation-algorithm-differential-evolution.1.

Botts, C. H. (2021). A novel metric for detecting anomalous ship behavior using a variation of the DBSCAN clustering algorithm. SN Computer Science, 2(5), 1–22. https://doi.org/10.1007/s42979-021-00804-4

Chahal, J. K., & Kaur, A. (2016). A hybrid approach based on classification and clustering for intrusion detection system. International Journal of Mathematical Sciences and Computing, 2(4), 34–40. https://doi.org/10.5815/ijmsc.2016.04.04

Chen, Z., & Li, Y. F. (2011). Anomaly detection based on enhanced DBScan algorithm. Procedia Engineering, 15, 178–182. https://doi.org/10.1016/j.proeng.2011.08.036

Diphoko, K. K. (2024). Introduction to machine learning-based intrusion detection systems (ML-IDS).

Dong, G., Jin, Y., Wang, S., Li, W., Tao, Z., & Guo, S. (2019). DB-Kmeans: An intrusion detection algorithm based on DBSCAN and K-means. In 2019 20th Asia-Pacific Network Operations and Management Symposium: Management in a Cyber-Physical World, APNOMS 2019. https://doi.org/10.23919/APNOMS.2019.8892910

Foster, M., Ronnqvist, H., & Fell, R. (2018). The performance of embankment dams with filters coarser than no-erosion design criteria. Hydropower & Dams, 4, 64–71.

Hasan, Y. (2024). Silhouette score and Davies-Bouldin index measurements on K-Means and DBSCAN cluster results. Journal of Applied Electrical Informatics and Engineering, 12(3S1), 60–74. https://doi.org/10.23960/jitet.v12i3s1.5001

Ibadirachman, R. K., Chrisnanto, Y. H., & Sabrina, P. N. (2024). DBSCAN parameter optimization uses the differential evolution method for anomaly detection in bank transaction data. Journal of Integrated Technology, 10(1), 22–31. https://doi.org/10.54914/jtt.v10i1.1189

Karczmarek, P., Kiersztyn, A., Pedrycz, W., & Al, E. (2020). K-Means-based isolation forest. Knowledge-Based Systems, 195, 105659. https://doi.org/10.1016/j.knosys.2020.105659

Laskar, M. T. R., Huang, J. X., Smetana, V., Stewart, C., Pouw, K., An, A., Chan, S., & Liu, L. (2021). Extending isolation forest for anomaly detection in big data via K-means. ACM Transactions on Cyber-Physical Systems, 5(4). https://doi.org/10.1145/3460976

Liputan6.com. (2021). Global financial services sector loses USD 100 billion annually due to cyber attacks.

Mboweni, I. V., Ramotsoela, D. T., & Abu-Mahfouz, A. M. (2023). Hydraulic data preprocessing for machine learning-based intrusion detection in cyber-physical systems. Mathematics, 11(8). https://doi.org/10.3390/math11081846

Rashid, U., Saleem, M. F., Rasool, S., Abdullah, A., & Mustafa, H. (2024). [Detail artikel tidak lengkap].

Schummer, P., del Rio, A., Serrano, J., Jimenez, D., Sánchez, G., & Llorente, A. (2024). Machine learning-based network anomaly detection: Design, implementation, and evaluation. AI (Switzerland), 5(4), 2967–2983. https://doi.org/10.3390/ai5040143

Setiawan, S. (2020). Talking about precision, recall, and F1-score.

Shim, J. S. (2021). Machine learning techniques for network analysis. Public Health, 125.

Shorewala, V. (2021). Anomaly detection and improvement of clusters using enhanced K-Means algorithm. In 2021 5th International Conference on Computer, Communication, and Signal Processing (ICCCS 2021) (pp. 115–121). https://doi.org/10.1109/ICCCSP52374.2021.9465539

Sikos, L. F. (2020). Packet analysis for network forensics: A comprehensive survey. Forensic Science International: Digital Investigation, 32, 200892. https://doi.org/10.1016/j.fsidi.2019.200892

Yuan, Y., Huang, Y., Yuan, Y., & Wang, J. (2024). Dynamic threshold-based two-layer online unsupervised anomaly detector. arXiv. http://arxiv.org/abs/2410.22967

Zamanian, Z., Feizollah, A., Anuar, N. B., Binti, L., Kiah, M., Technology, I., & Lumpur, K. (2017). Anomaly detection in policy authorization activity logs, 4(11), 383–388.