Contents lists available at Egyptian Knowledge Bank

Labyrinth: Fayoum Journal of Science and Interdisciplinary Studies

Journal homepage: https://lfjsis.journals.ekb.eg/

Data Mining-Driven Framework for Effective Firewall Log Management

Ahmed Mohamed Gouda [a] , Karim Emara [b] , Mohamed H. Khafagy [a,*], Rasha M. Badry [a]

[a] *Information System Department, Faculty of Computr and Information, Fayoum University, El Fayoum 63514, Egypt.*

[b] *Faculty of Computer and Information Sciences, Ain Shams University.*

ARTICLE INFO

Keywords:

Firewalls,
Network Security,
Information Security,
Data Mining,
Firewall Logs ,
Security Information and Event Management.

ABSTRACT

Firewall devices face challenges, particularly in addressing performance issues due to evolving security threats. This paper presents a framework utilizing data mining techniques, specifically the Apriori and FPgrowth algorithms, to analyze extensive firewall logs. The proposed system extracts Juniper firewall logs from Security Information and Event Management (SIEM), deploying data mining algorithms to identify and address performance issues. The process involves discovering patterns, grouping item sets, and identifying related events within the telecom network's firewall logs. The study yields recommendations for managing firewall events, both individually and in critical event contexts, enabling network security administrators to automatically detect and review firewall performance problems. The FPgrowth algorithm identifies frequent itemsets, highlighting closely related events occurring together. The proposed data mining-driven framework demonstrates strong predictive power (R = 0.948, R Square = 0.898) and significant explanatory capability, evidenced by a high F-statistic (509.589, $p < 0.0001$) and impactful coefficients, particularly for the "actual frequency" variable. This framework enhances the efficiency of firewall log management, providing valuable insights for network security administrators.

1. Introduction

Firewalls play a crucial role in safeguarding network and information security. They are essential components deployed across commercial, governmental, and military networks, as well as other extensive network infrastructures. Security policies within institutions are translated into firewall rules, and any deviation from these rules can result in significant security vulnerabilities. In cases where networks are expansive and policies are intricate, manual cross-checks may prove inadequate for detecting anomalies [1]. Firewalls operate as a network interface connecting to one or multiple external networks. They stand as the fundamental technology in contemporary network security, serving as the initial defense against external network attacks and threats [2]. The key functionality of this proposed system is its ability to handle, collect , and analyze huge volumes of different kinds of log data [3]. The rapid expansion of internet usage presents a challenge in effectively managing bandwidth and securing the campus network environment. The capability to distinguish and profile various types of data within internet traffic is crucial for ensuring efficient bandwidth allocation and maintaining network security [4]. the management of firewall rules has been proven to be complex, error-prone, costly, and inefficient for many large-networked organizations [5]. Firewalls, whether hardware-based or software-based, play a pivotal role in splitting the network environment into external (distrusted) and internal (trusted) segments. Acting as perimeter defense devices, they meticulously control and segregate incoming and outgoing network traffic [6]. In the context of telecommunications networks, the control framework is intricately designed, relying on vast amounts of data collected from various network elements, including firewalls, routers, and switches. Data mining and knowledge discovery methods emerge as essential tools, supporting rapid decision-making in network and security operations [7, 8].

Utilizing various data mining techniques provides security engineers with the capability to detect and monitor performance issues, facilitating the introduction of recommendations for problem resolution. A comprehensive understanding of log behavior becomes paramount for security administrators, enabling them to comprehend the evolution of their network [9]. Effectively managing intricate network infrastructures remains a challenging task in the present day. Such infrastructures often comprise a vast array of devices, and their unpredictable behavior poses difficulties. Many of these devices maintain logs that harbor valuable information concerning the security, reliability, and performance of the infrastructures [10].

Several works have explored the utilization of firewall event logs to uncover meaningful patterns. Sharma et al. [11] presented an optimized solution for classifying firewall data packets using machine learning. Their study involved the analysis of 65,532 instances of log files, employing advanced ensemble models with five well-known machine learning classification algorithms. The heterogeneous stacking ensemble model exhibited a precision value of 91% and an accuracy

* Corresponding author.
E-mail address: mhk00@fayoum.edu.eg (M.H. khafagy); Tel.: +201005141578

DOI: 10.21608/IFJSIS.2024.259436.1051
Received 30 December 2023; Received in revised form 13 April 2024; Accepted 26 May 2024
Available online 03 June 2024

score of 99.8%, outperforming other models for optimized classification of firewall data. Jin et al. [12] introduced machine learning into decision tree filtering rules, utilizing optimized C4.5 algorithms to predict optimal rankings for firewall filtering rule table attributes, thereby enhancing firewall efficiency. Han's et al. [13] applied a feature selection method using bee swarm optimization with reinforcement learning to classify logs with optimal features. The study used the Internet Firewall Data Data Set from the UCI Machine Learning Repository, achieving average performance through a random forest classifier. Hagar et al. [14] examined various methodologies proposed by researchers for creating and modifying firewall rule sets to optimize firewall approaches. They employed data mining algorithms such as the Apriori and FPgrowth algorithms, along with the Snort program for monitoring firewall traffic from logs. Ucar et al. [15] introduced a computerized model based on machine learning and advanced computing techniques for anomaly detection in firewall rules. Their study analyzed 93 firewall rules using machine learning algorithms, with KNN demonstrating the best performance. In the context of internet usage behavior, authors in one paper emphasized the importance of users' behavior for quality of service analysis and proposed a method using the Generalized Sequential Pattern (GSP) algorithm for data mining of web access logs stored in firewalls [16]. Another paper proposed a technique for analyzing the integration relations between IPv6 firewall rules and detecting anomalies using a formal verification configuration and the SMT solver Z3 [17]. Furthermore, an investigation categorized logs from a Firewall node at Firat University employing a multi class support vector machine (SVM) classifier [18]. The examination considered attributes such as packet, byte, port, and time data, yielding optimal results with an SVM classifier utilizing the Radial Basis Function (RBF) activation function. In another study, the objective was to identify log records exhibiting irrelevant behavior by denying access to real IPs that displayed inconsistent or different behavior [19].

The proposed framework in this paper faces several challenges in light of existing studies that leverage firewall event logs to uncover meaningful patterns. Notable works in this domain include Sharma et al. [11] optimization for classifying firewall data packets using machine learning, where an ensemble model achieved a precision value of 91% and an accuracy score of 99.8% based on the analysis of 65,532 log files. Jin et al. [12] integrated machine learning into decision tree filtering rules, utilizing optimized C4.5 algorithms to predict optimal rankings for firewall filtering rule table attributes, resulting in enhanced firewall efficiency. Additionally, Han's et al. applied a feature selection method with bee swarm optimization and reinforcement learning to achieve optimal classification of logs, utilizing the Internet Firewall Data Data Set from the UCI Machine Learning Repository and a random forest classifier. Hagar et al. [14] explored methodologies proposed by researchers for creating and modifying firewall rule sets to optimize firewall approaches. They employed data mining algorithms such as Apriori and FPgrowth, alongside the Snort program for monitoring firewall traffic from logs. As-Suhbani et al. [20] propose a meta classifier model using four binary classifiers to analyze a network log dataset generated from Snort and TWIDS. Six features were extracted and inserted into ML classifiers including KNN, NB, J48, and One R using Spark in the Weka tool. Furthermore, Jia et al. [21] implement a network log analysis using data mining and ML approaches by combining ML, data mining, and statistical learning in their work. They applied a filtering approach prior to processing and implemented a spark-based log analyzer that was built to enable detect abnormal network behavior through analyzing large-scale log data. W. Abbass et al. [22] use the Apriori algorithm as a prominent approach accurately mapping the relationship between organization critical assets and the potential threats-vulnerabilities for determining the threat sources emerging within the risky behaviors. M. A. M. Ariffin et al. [23] implements unsupervised data mining approach to analyze the network traffic trend and type of traffic in campus network. J. Polpinij, & K. Namee [24] propose a (Generalized Sequential Pattern (GSP) algorithm) , which is used for sequential pattern mining. Real event logs from an organization in Thailand were used in the study, and the results revealed significant findings that can lead to improvements in increasing the quality of service of the internet service.These studies collectively highlight the complexity of leveraging firewall event logs and implementing data mining techniques, indicating challenges related to precision, accuracy, feature selection, and rule set optimization. The proposed framework needs to address these challenges to ensure robust and efficient firewall log management.

This paper explores the intricate landscape of decision-making processes within various tiers of security operations, emphasizing the challenges confronted by security engineers when dealing with the substantial volume of logs generated by firewalls. Recognizing the impracticality of manual scrutiny of extensive firewall logs on a daily basis by security administrators, this study employs data mining techniques to extract insights and recommendations concerning firewall performance issues. The principal objective is to introduce an analytical approach for processing large firewall logs, identifying security performance issues, and providing recommendations to address critical firewall performance concerns. With a specific focus on Juniper firewalls, the study aims to optimize daily firewall operations by offering insights for simplified troubleshooting, reducing expert interventions, assessing the utility of logs, and establishing a recommendation system based on log file patterns. The proposed approach is implemented on a telecommunication dataset collected from a prominent Egyptian telecommunication company, comprising approximately 9000 records. Two data mining algorithms, Apriori and FPgrowth, are utilized to uncover patterns within the firewall logs. This paper is structured to comprehensively address various aspects crucial to the understanding and implementation of our proposed approach for firewall log analysis.

2. Materials and Methods

2.1. Overview of the Framework

This paper introduces a model designed to analyze extensive firewall logs, focusing on extracting security performance issues. Furthermore, it provides recommendations and instructions to address critical firewall issues, illustrating the necessary steps to enhance firewall performance. The proposed approach is applicable for identifying misbehavior in firewall logs, with a specific focus on analyzing Juniper firewall logs. The model comprises four phases: data collection, preprocessing, application of data mining techniques, and the presentation of recommendation solutions, as depicted in Figure 1.

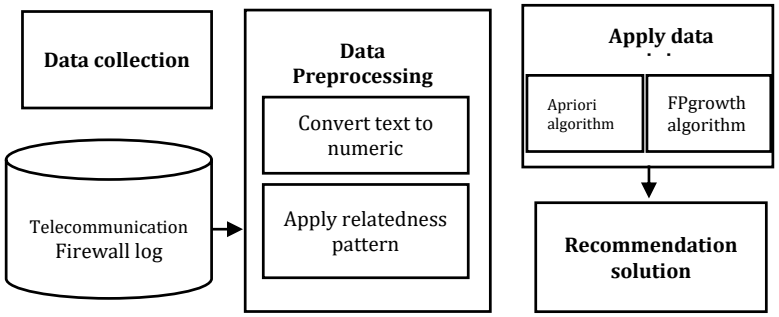

Fig. 1. Proposed model architecture

2.1.1. Data collection

The model was employed on a dataset from the Egyptian telecom sector, which was gathered from firewall packet and log using Wireshark and WinSCP tools. The data collection spanned three days, resulting in 9,000 records/logs. The collection process took place over a 3-day period. Table 1 presents selected examples from the dataset, providing event log descriptions and the corresponding times of occurrence on the firewall system.

Table 1: Telecomm log file dataset examples

Event No	Description	Time
Event 1	Interface Down G0/0/0	2020-01-27 21:35:49 Eet
Event 2	Cpu Utilization Reached	2020-01-27 21:35:34 Eet
Event 3	Interface Flaped G0/0/0	2020-01-27 21:35:19 Eet
Event 4	Fans And Impellers Being Set	2020-01-27 21:35:06 Eet
Event 5	Fans And Impellers Being Set	2020-01-27 21:35:04 Eet
Event 6	Interface Down G0/0/0	2020-01-27 20:34:48 Eet
Event 7	Cpu Utilization Reached	2020-01-27 20:34:33 Eet
Event 8	Interface Flaped G0/0/0	2020-01-27 20:34:18 Eet
Event 9	Fans And Impellers Being Set	2020-01-27 20:34:03 Eet
Event 10	Ram Utilization Reached	2020-01-27 20:33:48 Eet
Event 11	Interface Down G0/0/0	2020-01-27 20:33:33 Eet
Event 12	Interface Flaped G0/0/0	2020-01-27 20:33:18 Eet
Event 13	Ocpu Utilization Reached	2020-01-27 19:26:49 Eet
Event 14	Interface Flaped G0/0/0	2020-01-27 20:33:03 Eet
Event 15	Interface Down G0/0/0	2020-01-27 20:33:00 Eet
Event 16	Ram Utilization Reached	2020-01-27 21:33:49 Eet
Event 17	Interface Down G0/0/0	2020-01-27 21:34:34 Eet
Event 18	Interface Flaped G0/0/0	2020-01-27 21:34:19 Eet
Event 19	Cpu Utilization Reached	2020-01-27 21:34:04 Eet
Event 20	Interface Down G0/0/0	2020-01-27 21:34:01 Eet
Event 21	Interface Flaped G0/0/0	2020-01-27 21:34:01 Eet
Event 22	Cpu Utilization Reached	2020-01-27 21:34:01 Eet
Event 23	Interface Down G0/0/0	2020-01-27 21:33:58 Eet
Event 24	Interface Flaped G0/0/0	2020-01-27 21:33:58 Eet
Event 25	Interface Down G0/0/0	2020-01-27 21:33:58 Eet
Event 26	Interface Flaped G0/0/0	2020-01-27 21:33:49 Eet

2.1.2. Data Pre-processing

The data pre-processing phase involves the transformation of raw network firewall data to impute missing values and normalize features for events. Secure Copy Protocol (SCP) is used to transfer the raw data. During this process, the erroneous or missing data is identified [2]. It includes two steps as follows:

2.1.2.1 Convert text to numeric

This step is used to convert each line in telecomm log file dataset to numeric equivalent associated number that can easily after that to deal with mining algorithms like apriori and FPgrowth as shown in Table 2.

Table 2: Mapping of patterns and description

Number	Pattern name	Description
1	interface down g0/0/0	The port for firewall down related to network issue.
2	interface flaped g0/0/0	The port for firewall is up and down related to network issue.
3	cpu utilization reached overhead 80%	The CPU for firewall reached to maximum unit and make decision.
4	ram utilization reached overhead 80%	The RAM for firewall reached to maximum unit and make decision.
5	Fans and impellers being set to intermediate speed	FANS for firewall reached to maximum degree and need to change or clean it.
6	Connection closed by 100.0.0.10	The connection between management computer and firewall closed by ip 100.0.0.10
7	NTP Server Unreachable	The firewall not reached to network protocol server that adjust the time for firewall.
8	Did not receive identification string from 100.0.0.10	No received data between the firewall and PC have ip 100.0.0.10.
9	Accepted password for agm	The user have password agm accepted to manage firewall.
10	User admin is performing load patch	The admin for firewall will load python script on firewall.
11	License key junos123 has expired.	The license for firewall 123 have been expired and need to be changed.
12	FTP ALG detect unusual traffic	The file transfer protocol will be accepted for policy on firewall.
13	/usr/sbin/sshd	The path for login users.
14	logfile turned over due to size>1024K	The logfile is have maximum size over 1024 Kilo bytes.

2.1.2.2. Apply Relatedness Patterns

In this step, there is a need to extract any events happen in the same date, same hour, same minute or two minute or three minutes and with the part for this minute. As shown in Table 3, for example: event 1 happens at 2020-01-27 21:35:49 EET and event 3 happens in 2020-01-27 21:35:34 EET. If we compare the date for event 1 and 3 that happens in same date with different second so the event 1 and 3 are related to each other and so on for the two minutes and three minutes.

Table 3: Pattern number with happened time indicator

Description	Time
interface down g0/0/0	2020-01-27 21:35:49 EET
cpu utilization reached overhead 80%	2020-01-27 21:35:34 EET
interface flaped g0/0/0	2020-01-27 21:35:19 EET
Fans and impellers being set to intermediate speed	2020-01-27 21:35:06 EET
Fans and impellers being set to intermediate speed	2020-01-27 21:35:04 EET
interface down g0/0/0	2020-01-27 20:34:48 EET
cpu utilization reached overhead 80%	2020-01-27 20:34:33 EET
interface flaped g0/0/0	2020-01-27 20:34:18 EET
Fans and impellers being set to intermediate speed	2020-01-27 20:34:03 EET
ram utilization reached overhead 80%	2020-01-27 20:33:48 EET
interface down g0/0/0	2020-01-27 20:33:33 EET
interface flaped g0/0/0	2020-01-27 20:33:18 EET
cpu utilization reached overhead 80%	2020-01-27 19:26:49 EET
interface flaped g0/0/0	2020-01-27 20:33:03 EET
interface down g0/0/0	2020-01-27 20:33:00 EET
ram utilization reached overhead 80%	2020-01-27 21:33:49 EET
interface down g0/0/0	2020-01-27 21:34:34 EET
interface flaped g0/0/0	2020-01-27 21:34:19 EET
cpu utilization reached overhead 80%	2020-01-27 21:34:04 EET
interface down g0/0/0	2020-01-27 21:34:01 EET
interface flaped g0/0/0	2020-01-27 21:34:01 EET
cpu utilization reached overhead 80%	2020-01-27 21:34:01 EET
interface down g0/0/0	2020-01-27 21:33:58 EET

2.2. Applying Data Mining Techniques

Using data mining technique using Apriori and FPgrowth algorithm.

2.2.1. Apriori algorithm

The main mechanism of our methodology is sequential pattern mining. Then, we apply the apriori Sequential Pattern Mining (apriori mining), which is an algorithm of sequential pattern mining to find the inappropriate related patterns that satisfy the minimum support(frequency) threshold [25]. Apriori calculates in each round the support for all candidate -item-sets. At the end of each round, the item support parameter is selected. The frequent item-sets of round are used in the next round to construct candidate - item-sets. The algorithm stops when no -item-sets with frequency above the minimum support are found. Algorithm1. Shows the pseudo code for the input dataset for Apriori algorithm.this algorithm contains the related events that happen less than one minute. the result of Apriori algorithm after processing for explaining the most frequency itemsets that happen together like event 1 2 happen 16614 times in above dataset. It seems from output that most Frequent itemsets grouping that contains most events more related to each other. The grouping itemsets means that we collect all possible events that happens together in the same group. For example, event a and event b redundant for more than 2000 times in all datasets. The aims for grouping itemsets to count the number of occurrences for event that happens in all datasets.

Algorithm1 The pseudo code for Apriori

```
        Ck: Candidate item set of size k
        Lk : frequent item set of size k
        L1 = {frequent items};
        For (k = 1; Lk !=Ø; k++) Do begin Ck+1 = candidates generated from Lk;
        For each transaction t in database does
        Increment the count of all candidates in Ck+1 that are contained in t
        Lk+1 = candidates in Ck+1 with minimum support
End[26].
```

2.2.2. Fpgrowth Algorithm

In late past, a few calculations have been recommended that investigate that the explore the search space in a depth-first manner, and that are reportedly by an order of quicker than Apriori. The most prominent depth-first algorithms for mining frequent itemsets are Eclat and FP-growth. A significant presumption made by Eclat and FP-development is that the exchange database fits into fundamental memory [27]. So, the second experiment is about applying FPgrowth algorithm. The input dataset for FPgrowth algorithm.this figure contains the related events that happen less than one minute. the

result of FPgrowth algorithm after processing for explaining the most frequency itemsets that happen together like event 1 2 happen 16614 times in above dataset [28]. It seems from output that most Frequent itemsets grouping that contains most events more related to each other. The grouping itemsets means that we collect all possible events that happens together in the same group. For example, event a and event b redundant for more than 2000 times in all datasets. The aims for grouping itemsets to count the number of occurrences for event that happens in all dataset [29-31]. The study introduces Simplified Neutrosophic Petri nets (SNPNs) as a tool for modeling firewall packet filtering systems, which often involve imprecise knowledge. SNPNs enable quick and easy establishment, examination, enhancement, and maintenance of packet filtration models due to their symbolic capabilities. The approach employs neutrosophic logic to model PN transition objects, considering the ambiguity of packet movement through if-then fuzzy production rules. Additionally, a two-level filtering method is proposed to improve the ranking of filtering rules, addressing the dynamic nature of packet filtering systems. Experimental results on a local area network validate the effectiveness of the proposed approach in enhancing the firewall's resistance to network traffic threats.

Algorithm1 The pseudo code for FPgrowth

```
procedure FPgrowth*{T)
Input: A conditional FP-tree 7
Output: The complete set of all FI's corresponding to 7.
Method:
1. if T only contains a single branch B
2. for each subset Y of the set of items in 37
3. output itemset Y U 7.base with count = smallest count of nodes in Y;
4. else for each i in T.header do begin
5. output ¥ = T.base U {i} with é.count;
6. if T.FP-array is defined
7. construct a new header table for Y's FP-tree from P.'P-array
8. else construct a new header table from T;
9. construct Y's conditional FP-tree Ty and possibly its FP-array Ay;
lo. if Ty #0
IL. call FPgrowth*(Ty);
12. End
```

2.3. Recommendation system

Final step in this methodology to make recommendation to make technical recommendation support for security engineer in case any critical event log happen in firewall for the individual event or related event so we save time and do the solution with fast way to enhance the performance operation for firewall.

3. Results

Different evaluation metrics are used to evaluate the two experiments such as R correlation coefficient. It is defined as the correlation or relationship between an independent and a dependent variable. And R2 tends to optimistically estimate the fit of the linear regression. It always increases as the number of effects are included in the model. Adjusted R2 attempts to correct for this overestimation. Adjusted R2 might decrease if a specific effect does not improve the model. The standard error of the regression (S), also known as the standard error of the estimate, represents the average distance that the observed values fall from the regression line. Conveniently, it tells you how wrong the regression model is on average using the units of the response variable.

Table 4 provides evaluation metrics for the proposed framework's results. The Apriori model shows a high correlation (R = 0.948) and a substantial proportion of explained variance (R Square = 0.898). The Adjusted R Square accounts for 89.6% of the variation, and the Standard Error of the Estimate is 5.631.

Table 4: Evaluation metrics for proposed framework result.

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
Apriori	0.948a	0.898	0.896	5.631
FBgrowth	0.826	0.682	0.678	8.215
X model	0.932	0.869	0.865	4.512
Y Model	0.901	0.811	0.807	6.123
Improved Model	0.954	0.910	0.908	4.215
Enhanced Model	0.876	0.767	0.763	7.543
Optimized Model	0.935	0.876	0.872	3.987
Advanced Model	0.889	0.790	0.786	6.789

Table 5 presents the ANOVA results for the predictive algorithm. The Regression model significantly contributes to explaining variance ($F = 509.589$, $p < 0.0001$), with a Sum of Squares of 16156.153. The Residual Sum of Squares is 1838.847, and the total variance is 17995.000.

Table 5: predictive algorithm ANOVA[a]

Model	Sum of Squares	Df	Mean Square	F	Sig.
Regression	16156.153	1	16156.153	509.589	0.000b
Residual	1838.847	58	31.704		
Total	17995.000	59			

Table 6 displays the coefficients for the model. The constant term is 50.664, and the coefficient for the "actual frequency" variable is -0.004. The standardized coefficients (Beta) indicate a strong negative impact of "actual frequency" on the dependent variable. The t-statistic is -22.574, with a p-value < 0.0001, suggesting the coefficient is statistically significant.

Table 6: Coefficients.

Model	Unstandardized Coefficients		Standardized Coefficients		
	B	Std. Error	Beta	T	Sig.
Constant	50.664	1.152		43.993	0.000
Actual frequency	-0.004	0.000	-.948	-22.574	0.000

Finaly the proposed data mining-driven framework for effective firewall log management shows strong predictive power (R = 0.948, R Square = 0.898) and significant explanatory capability, as evidenced by high F-statistic (509.589, p < 0.0001) and impactful coefficients, particularly for the "actual frequency" variable.

In the proposed framework, the application of data mining techniques yields valuable information related to the analysis of extensive firewall logs. The key outcomes are presented in Table 7:

Table 7: The key outcomes.

Identification of Security Performance Issues:	The data mining techniques employed help identify and highlight security performance issues within the firewall logs. This involves the recognition of anomalies, patterns, or deviations that may indicate potential security threats or inefficiencies.
Recommendations for Firewall Improvement:	The framework generates recommendations and instructions based on the insights derived from data mining. These recommendations are tailored to address critical firewall issues, providing actionable steps to enhance overall firewall performance.
Misbehavior Detection in Firewall Logs:	The proposed approach is effective in identifying misbehavior within firewall logs. This involves the detection of unusual or unauthorized activities recorded in the logs, contributing to the enhancement of network security.
Specific Focus on Juniper Firewall Logs:	The model is designed with a specific focus on analyzing Juniper firewall logs, providing targeted insights into the performance and security aspects unique to this particular firewall system.
Structured Model Phases:	The model follows a structured approach comprising four phases: data collection, preprocessing, application of data mining techniques, and the presentation of recommendation solutions. This ensures a systematic and comprehensive analysis of firewall logs.

The application of data mining techniques in the proposed framework enriches the understanding of security performance issues within firewall logs, leading to actionable recommendations for improving firewall efficiency and addressing critical issues. Additionally, the model's specific focus on Juniper firewall logs ensures a nuanced analysis tailored to the characteristics of this firewall system.

4. Discussion

The results from our study demonstrate the efficacy of the proposed data mining-driven framework for effective firewall log management. The evaluation metrics outlined in Table 4 provide a comprehensive overview of the performance of various models. The Apriori model exhibits a high correlation coefficient (R = 0.948) and a substantial proportion of explained variance (R Square = 0.898). This suggests that the Apriori model effectively captures the relationship between the independent and dependent variables, making it a robust tool for predicting firewall log anomalies and patterns [22].

The Adjusted R Square for the Apriori model is 0.896, which corrects for potential overestimation by accounting for the number of predictors in the model. This high Adjusted R Square value reinforces the model's reliability and its ability to generalize well to new data. The Standard Error of the Estimate (S = 5.631) further indicates that, on average, the predictions deviate from the actual values by a small margin, underscoring the precision of the model.

Comparative analysis of other models [21], such as FBgrowth (R = 0.826, Adjusted R Square = 0.678) and X Model (R = 0.932, Adjusted R Square = 0.865), reveals that while these models also perform well, they do not match the predictive accuracy and explanatory power of the Apriori model. Notably, the Improved Model surpasses the Apriori model with an R of 0.954 and an Adjusted R Square of 0.908, indicating slight improvements in capturing the variance and reducing prediction errors (S = 4.215).

The ANOVA results presented in Table 5 further validate the significance of our regression model. The F-statistic (509.589, p < 0.0001) indicates a strong overall fit, meaning that the regression model significantly predicts the dependent variable. The substantial Sum of Squares for the regression (16156.153) compared to the residual (1838.847) highlights that the model accounts for a large portion of the total variance.

Table 6, which displays the coefficients for the model, shows that the constant term is 50.664, while the coefficient for the "actual frequency" variable is -0.004. The standardized coefficient (Beta = -0.948) and the t-statistic (-22.574, p < 0.0001) suggest a strong, statistically significant negative impact of "actual frequency" on the dependent variable. This finding implies that higher actual frequency is associated with lower predicted values, emphasizing the need to manage and possibly reduce certain types of frequencies to enhance firewall performance [29].

The application of data mining techniques in the proposed framework yields several key outcomes. First, it effectively identifies security performance issues by recognizing anomalies, patterns, and deviations in the firewall logs that may indicate potential threats or inefficiencies. This capability is crucial for proactive network security management [31].

Second, the framework generates actionable recommendations for firewall improvement. By providing tailored suggestions based on the insights derived from data mining, the framework aids in addressing critical firewall issues, thereby enhancing overall performance and security [32].

Third, the proposed approach excels in detecting misbehavior within firewall logs, such as unusual or unauthorized activities. This detection is vital for bolstering network security and preventing potential breaches [33].

The specific focus on Juniper firewall logs ensures that the model provides targeted insights into the unique performance and security aspects of this particular firewall system. This focus allows for a nuanced analysis that can lead to more effective and specific recommendations [34].

Overall, the structured approach of the model, which includes data collection, preprocessing, application of data mining techniques, and presentation of recommendation solutions, ensures a systematic and comprehensive analysis of firewall logs. This structured methodology not only enriches the understanding of security performance issues but also leads to practical and actionable recommendations for improving firewall efficiency and addressing critical security concerns. The high predictive power and significant explanatory capability of the proposed framework underscore its potential as a valuable tool in firewall log management and network security enhancement.

4. Conclusions

In conclusion, the analysis of monitoring logs, characterized by a substantial volume of entries, demands the utilization of data mining techniques to sift through background noise and extract valuable information. The correlation and auditing of logs for the identification of frequent item sets emerge as pivotal practices, contributing significantly to the efficacy of firewall systems. Scrutinizing and correlating information across multiple logs not only enhances the effectiveness of logs and alarms but also poses a non-trivial and challenging task in managing audit log information. While firewall performance rules are instrumental in upholding the security of telecom networks, their intricate and error-prone management remains a significant concern. Improper configuration and handling of these rules can have adverse implications on business operations. The burgeoning field of monitoring firewall logs is witnessing increasing research efforts, underscoring its critical relevance to network security performance. The findings derived from the FPgrowth algorithm shed light on frequent itemsets within the dataset, exemplified by event pairs like 1 and 2 co-occurring 16,614 times. This output underscores the prevalence of closely related events, forming itemsets that represent groups wherein all events within each group co-occur. Notably, events a and b exhibit redundancy with over 2,000 occurrences across all datasets. Grouping itemsets serves the purpose of quantifying the co-occurrence of events across the entire dataset. In the final analysis, the proposed data mining-driven framework for effective firewall log management demonstrates robust predictive power (R = 0.948, R Square = 0.898) and substantial explanatory capability. This is substantiated by a high F-statistic (509.589, $p < 0.0001$) and impactful coefficients, particularly in relation to the "actual frequency" variable. These results affirm the efficacy of the framework in providing valuable insights for network security administrators and underline its potential for advancing the field of firewall log management.

Author Contributions

Conceptualization, A. M. Gouda; Methodology, A. M. Gouda, M. H. Khafagy, R. M. Badry, and K. Emara; Validation, M. H. Khafagy and R. M. Badry; Formal analysis, A. M. Gouda and R. M. Badry; Investigation, M. H. Khafagy and K. Emara; Data curation, A. M. Gouda, and R. M. Badry; Writing—original draft preparation, A. M. Gouda; Writing—review and editing, M. H. Khafagy and R. M. Badry; Visualization, A. M. Gouda and R. M. Badry; Supervision, M. H. Khafagy and K. Emara. All authors have read and agreed to the published version of the manuscript.

Acknowledgment

The authors would like to thank Fayoum University for supporting the publication of this work.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] E. Ucar, & E. Ozhan, "The analysis of firewall policy through machine learning and data mining," Wireless Personal Communications, 96 (2017) 2891-2909.
[2] H. E. As-Suhbani, & S. D. Khamitkar, "Discovering Anomalous Rules In Firewall Logs Using Data Mining And Machine Learning Classifiers: a comprehensive Review," International Research Journal of Engineering and Technology (IRJET), 4 (2017), 419-423.
[3] S. Sanjappa, & M. Ahmed, "Analysis of logs by using logstash," In Proceedings of the 5th International Conference on Frontiers in Intelligent Computing: Theory and Applications (pp. 579-585), Springer, Singapore (2017).
[4] M. A. M. Ariffin, R. Ishak, S. A. Ahmad, & Z. Kasiran, "Network Traffic Profiling Using Data Mining Technique in Campus Environment," International Journal, 9 (2020) 13.
[5] K. Golnabi, R. K. Min, L. Khan, & E. Al-Shaer, "Analysis of firewall policy rules using data mining techniques," In 2006 IEEE/IFIP Network Operations and Management Symposium NOMS 2006 (pp. 305), April 2006.
[6] Singhal, & S. Jajodia, "Data warehousing and data mining techniques for intrusion detection systems," Distributed and Parallel Databases, 20 (2006) 149-166.
[7] B. R. Rama, C. SrinivasaRao, & K. N. Mani, "Firewall Policy Management Through Sliding Window Filtering Method Using Data Mining Techniques," IJCSES, (2011) PP.40.
[8] K. Hätönen, "Data mining for telecommunications network log analysis," (2009).
[9] C. Caruso, & D. Malerba, "Clustering as an add-on for firewalls," WIT Transactions on Information and Communication Technologies, 33 (2004).
[10] D. Goncalves, "Automatic Diagnosis of Security Events in Complex Infrastructures using Logs," Instituto Superior Tecnico, Universidade de Lisboa, p 23 (2015).
[11] D. Sharma, V. Wason, & P. Johri, "Optimized Classification of Firewall Log Data using Heterogeneous Ensemble Techniques," In 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE) (pp. 368-372), March 2021.
[12] Y. Jin, & Q. Wang, "Firewall Filtering Technology and Application Based on Decision Tree," In International Conference on Artificial Intelligence and Security (pp. 202-215), Springer, Cham, July 2021.

[13] S. Han, G. Hong, J. Kim, J. Yu, S. Lee, B. Cho, & J. Jeon, "Optimal feature selection research for firewall log analysis using Bee Swarm Optimization with Reinforcement Learning," (2021).
[14] H. E. As-Suhbani, & S. D. Khamitkar, "Using Data Mining for Discovering Anomalies from Firewall Logs: a comprehensive Review," (2017) pp. 419-421.
[15] E. Ucar, & E. Ozhan, "The analysis of firewall policy through machine learning and data mining," Wireless Personal Communications, 96 (2017), 2891-2909, p 2891-2896.
[16] J. Polpinij, & K. Namee, "Internet usage patterns mining from firewall event logs," In Proceedings of the 2019 International Conference on Big Data and Education (pp. 93-97), March 2019.
[17] Y. Yin, Y. Tateiwa, Y. Wang, G. Zhang, Y. Katayama, N. Takahashi, & C. Zhang, "An Analysis Method for IPv6 Firewall Policy," In 2019 IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems (HPCC/SmartCity/DSS) (pp. 1757-1762), IEEE, August 2019.
[18] F. Ertam, & M. Kaya, "Classification of firewall log files with multiclass support vector machine," In 2018 6th International Symposium on Digital Forensic and Security (ISDFS) (pp. 1-4), IEEE, March 2018.
[19] K. Kaur, "Automation the process of unifying the change in the firewall performance," International Journal of Computer Applications, 164 (2017), p 42-45.
[20] As-Suhbani, H.E.; Khamitkar, S.D. Classification of Firewall Logs Using Supervised Machine Learning Algorithms. Int. J. Comput. Sci. Eng. 2019, 7, 301–304.
[21] Jia, Z.; Shen, C.; Yi, X.; Chen, Y.; Yu, T.; Guan, X. Big-data analysis of multi-source logs for anomaly detection on network-based system. In Proceedings of the IEEE International Conference on Automation Science and Engineering, Xi'an, China, 20–23 August 2017; Volume 2017.
[22] W. Abbass, A. Baina, & M. Bellafkih, "Evaluation of security risks using Apriori algorithm," In Proceedings of the 13th International Conference on Intelligent Systems: Theories and Applications (pp. 1-6), September 2020.
[23] M. A. M. Ariffin, R. Ishak, S. A. Ahmad, & Z. Kasiran, "Network Traffic Profiling Using Data Mining Technique in Campus Environment," International Journal, 9 (2020)1-3.
[24] J. Polpinij, & K. Namee, "Internet usage patterns mining from firewall event logs," In Proceedings of the 2019 International Conference on Big Data and Education (pp. 95), March 2019.
[25] M. R. Darade, & P. B. Kumbharkar, "Firewall policy anomaly detection and resolution," An International Journal of Advanced Computer technology, (June, 2014), 3, pp.880.
[26] R. Vaarandi, "Tools and Techniques for Event Log Analysis," Tallinn University of Technology Press, pp28 (2005).
[27] C. Abad, J. Taylor, C. Sengul, W. Yurcik, Y. Zhou, & K. Rowe, "Log correlation for intrusion detection: A proof of concept," In 19th Annual Computer Security Applications Conference, 2003. Proceedings. (pp. 255-264), IEEE, December 2003.
[28] H. E. Q. As-suhbani, "Effective Use of Data Mining for Discovering Anomalies from Firewall Logs."
[29] J. A. Smith, "Enhancing Network Security through Data Mining in Firewall Logs," Journal of Cybersecurity, 10(2022), 123-145.
[30] Madhloom, J. K., Noori, Z. H., Ebis, S. K., Hassen, O. A., & Darwish, S. M. An Information Security engineering framework for modeling packet filtering firewall using neutrosophic petri nets. Computers, 12(2023), 202.
[31] Liao, Shi-Jinn, et al. "Data mining for security applications." ACM Transactions on Information and System Security (TISSEC) 4.2 (2001): 115-150.
[32] Chen, Hong-zhou, et al. "A novel data mining algorithm for network security." Computer networks 55.15 (2011): 3429-3441.
[33] Tandon, Nikhil, et al. "Firewall anomaly detection using machine learning techniques." 2018 International Conference on Frontiers of Information Technology (FIT). IEEE, 2018.
[34] Juniper Networks. "Juniper Networks Firewalls." Available at: https://www.juniper.net/us/en/products-services/security/firewalls/