إقـــــرار

أنا الموقع أدناه مقدم الرسالة التي تحمل العنوان:

A Model for Strengthening Accuracy in Detecting the Anomalous Firewall Rules in Small Network (SADAR)

نموذج لتعزيز دقة أمن الشبكات الصغيرة من خلال الكشف عن القواعد الشاذة في جدار الحماية

أقر بأن ما اشتملت عليه هذه الرسالة إنما هو نتاج جهدي الخاص، باستثناء ما تمت الإشارة إليه حيثما ورد، وإن هذه الرسالة ككل أو أي جزء منها لم يقدم من قبل لنيل درجة أو لقب علمي أو بحثي لدى أي مؤسسة تعليمية أو بحثية أخرى.

DECLARATION

The work provided in this thesis, unless otherwise referenced, is the researcher's own work, and has not been submitted elsewhere for any other degree or qualification

Student's name: Emad KH. M. Elrayyes

اسم الطالب: عماد خميس مدحت الريس

Signature:

التوقيع:

Date: 2015/07/11

التاريخ: 2015/07/11

The Islamic University of Gaza
Deanship of Graduate Studies
Faculty of Information technology

A Model for Strengthening Accuracy in Detecting the Anomalous Firewall Rules in Small Network (SADAR)

Prepared by
Emad KH. Elrayyes

Supervised by
Dr. Tawfiq SM. Barhoom

A Thesis Submitted as Partial Fulfillment of the Requirements for the Degree of Master in Information Technology

(2015AD, 1436AH)

بسم الله الرحمن الرحيم

الجامعة الإسلامية – غزة
The Islamic University - Gaza

مكتب نائب الرئيس للبحث العلمي والدراسات العليا هاتف داخلي: 1150

الرقم س.غ/35/..... Ref

التاريخ 2015/06/07 Date

نتيجة الحكم على أطروحة ماجستير

بناءً على موافقة شئون البحث العلمي والدراسات العليا بالجامعة الإسلامية بغزة على تشكيل لجنة الحكم على أطروحة الباحث/ عماد خميس مدحت الريس لنيل درجة الماجستير في كلية *تكنولوجيا المعلومات* برنامج تكنولوجيا المعلومات وموضوعها:

نموذج لتعزيز دقة أمن الشبكات الصغيرة من خلال الكشف عن القواعد الشاذة في جدار الحماية

A Model for Strengthening Accuracy in Detecting the Anomalous Firewall Rules in Small Network (SADAR)

وبعد المناقشة التي تمت اليوم الأحد 19 شعبان 1436هـ، الموافق 2015/06/07م الساعة العاشرة صباحاً، اجتمعت لجنة الحكم على الأطروحة والمكونة من:

د. توفيق سليمان برهوم	مشرفاً و رئيساً	.....
د. أشرف محمد العطار	مناقشاً داخلياً	.....
د. إبراهيم خليل برهوم	مناقشاً خارجياً	.....

وبعد المداولة أوصت اللجنة بمنح الباحث درجة الماجستير في كلية *تكنولوجيا المعلومات*/ برنامج تكنولوجيا المعلومات.

واللجنة إذ تمنحه هذه الدرجة فإنها توصيه بتقوى الله ولزوم طاعته وأن يسخر علمه في خدمة دينه ووطنه.

والله ولي التوفيق،،،

مساعد نائب الرئيس للبحث العلمي والدراسات العليا

أ.د. فؤاد علي العاجز

شئون البحث العلمي والدراسات العليا
الجامعة الإسلامية غزة
Research & Graduate Affairs

P.O. Box 108, Rimal, Gaza, Palestine fax: +970 (8) 286 0800 فاكس Tel: +970 (8) 286 0700 ص.ب. 108 الرمال، غزة، فلسطين هاتف
public@iugaza.edu.ps www.iugaza.edu.ps

Dedication

To the soul of my brother in his eternal existence

To My Beloveds

Father, Mother

Sisters and Brother, Wife and Children

Supervisor Dr. Tawfiq SM. Barhoom

Best Friends

Acknowledgment

All praise is to Allah

I would like to express my deep and sincere gratitude to my supervisor, Associate Professor: Tawfiq SM. Barhoom, Dean Faculty of Information Technology, in Islamic University-Gaza, for his guidance and for giving me the opportunity to accomplish this research. He advised me during my work to present the research as clearly as possible.

I would like to extend my thanks to the academic staff of the Faculty of Information Technology in Islamic University of Gaza who helped me during my master's study and taught me different courses.

I would like to extend my thanks to the ministry of communications and Information Technology – Palestine and network administrators Eng. Nedal Safady and Eng. Fouad Abu Aomir, In helping and supporting me in all phases of the experiments the "SADAR" model for the detecting the anomalous rules in information technology Labs in ministry.

Emad KH. Elrayyes

Table of Contents

List of Figures

List of Tables

List of Abbreviations

IP Internet Protocol.

TCP Transmission Control Protocol.

UDP User Datagram Protocol.

MS Milliseconds.

SADAR A Model for **S**trengthening **A**ccuracy in **D**etecting the **A**nomalous Firewall **R**ules in Small Network

Abstract

The firewall policy rules is a core technology that has an important role in the network security, through controlling the traffic with income and outgoing the packets over the network. Moreover, the management of the firewall policy rules is a very complicated function and error prone.

However, the poor management of the firewall policy rules work on vulnerability the network security and this is the main reasons to cause conflict between two rules or more. The conflict between the rules it called the anomalous firewall policy rules. There are five type of anomalous rules namely (shadowing anomalous, generalization anomalous, correlation anomalous, redundancy anomalous and irrelevance anomalous), each type of anomalous rules has a different degree of overlapping complexity between the rules.

We built a model for strengthening accuracy in detecting the anomalous firewall rules in a small network, supported in the detection four type of anomalous rules namely (shadowing, generalization, correlation and redundancy anomalous). We applied different mechanism in matching process, through divided the IP address to four segments in array and matching every element in segment position with other element segment position in the same position and use the subnet mask to select the size of IP range.

We applied sixteen different experiment with different dataset sizes in detection the anomalous rules, and we used the confusion matrix in evaluate the result according to overall accuracy, and was the average of previous experiments according to the overall accuracy is 92.71% . We believe that the result was acceptable because not there are any results in related research to compare with it.

Keywords: Accuracy Factor, Anomalous Rules, Firewall Policy Rules, Mismanagement.

عنوان البحث باللغة العربية

نموذج لتعزيز دقة أمن الشبكات الصغيرة من خلال الكشف عن القواعد الشاذة في جدار الحماية

ملخص:

تعتبر قواعد وسياسات جدار الحماية من التكنولوجيا الأساسية التي لها دور مهم في أمن الشبكات، من خلال السيطرة على حركة دخول وخروج حزم البيانات عبر الشبكة، غير أن إدارة قواعد جدار الحماية تعتبر وظيفة معقدة للغاية وعرضة للخطأ.

ومع ذلك، فإن سوء إدارة قواعد وسياسات جدار الحماية تعمل على هشاشة أمن الشبكات، وهذا يعتبر من الأسباب الرئيسية التي ينتج عنها تضارب بين قاعدتين أو أكثر في مراقبه حزم البيانات، وهذا التضارب يسمى قواعد جدار الحماية الشاذة، ويوجد هناك خمسة أنوع من القواعد الشاذة كما يلي (التظليل الشاذ، تعميم الشاذ، الارتباط والتكرار الشاذ وعديمة الاعتداد)، ولكل نوع من هذه القواعد الشاذة لديها سلوك مختلف في التضارب والتداخل.

ولذلك، فإننا بحاجة لتعزيز عامل الدقة والذي يعتبر أمر بالغ الأهمية في إدارة قواعد وسياسات جدار الحماية من خلال الكشف عن قواعد جدار الحماية الشاذة قبل أي إضافة أو حذف أو التحديث في قائمة قواعد جدار الحماية، وعلية قمنا ببناء نموذج لتعزيز دقة في الكشف عن الحالات الشاذة بقواعد جدار الحماية خاص بالشبكات الصغيرة، يعمل على الكشف عن أربعة أنوع من القواعد الشاذة (التظليل، التعميم، الارتباط والتكرار الشاذ).

ولتحقيق هدفنا الرئيسي من بحثنا في تعزيز عامل الدقة في الكشف عن القواعد الشاذة، ولقد تم استخدم آلية مختلفة في عملية الكشف من خلال تقسيم عنوان **IP** الى اربعة اقسام من كل عنوان يحتوي على **IP** بجميع القواعد ومطابقة كل عنصر الناتج من عملية التقسيم بالعنصر الذي يقابله بالإضافة الى استخدام قناع الشبكة لتحديد حجم **IP**.

لقد اجرينا ستة عشر من التجارب المختلفة مع أحجام مختلفة من بيانات في الكشف عن القواعد الشاذة، واستخدمنا مصفوفة الارتباك في تقييم النتيجة وفقا لمعايير الدقة الشاملة، وكان متوسط نتائج التجارب السابقة وفقا لمعيار الدقة الشاملة هي 92.71٪، ونحن نعتقد أن النتيجة كانت مقبولة وذلك لعدم عرض النتائج في الأبحاث الأخرى ذات العلاقة لمقارنتها معها.

الكلمات المفتاحية: عامل الدقة، قواعد وسياسات جدار حماية، القواعد الشاذة، سوء الإدارة

CHAPTER 1: Introduction.

This chapter provides a brief introduction for our thesis, which explains the underlying concepts of firewall policy rules and anomalous of firewall policy rules and it talks about the thesis problem, the research objectives, the research importance, the research scope and limitations, as well as the research methodology.

In [9] the firewalls are core elements in network security. However, managing firewall rules, especially for enterprise networks, has become very complex and error-prone. Firewall filtering rules have to be carefully managing and organized in order to correct the implement of the security Policy. In addition, inserting or modifying a filtering rule requires a thorough analysis of the relationship between this rule and other rules in order to determine the proper order of this rule and commit the updates.

The firewall policy rules have become important and core element in networks security, the firewalls have been the frontier defense for secure networks against attacks and unauthorized traffic by filtering out unwanted network traffic coming into or going from the secured network. The filtering decision is takes according to a set of ordered filtering rules defined based on predefined security policy requirements [17].

In [22] is defined the managing of firewall technology is core step toward networks secure, the complexity of modification and update firewall policy rules, might limit the effectiveness of firewall security, a firewall policy rules may include anomalous rules, where some a packet may match with two or more different filtering rules.

On other hand, when the filtering rules are define as a serious attention has to rule relations and interactions in order to determine the proper rule ordering and guarantee correct security policy semantics. With increasing the numbers of firewall policy rules, the complexity of writing a new rule or modifying an existing one also increases, It is very likely, in this case, result an anomalous policy rules such as, shadowing, generalization, correlation and redundancy Anomalous rules [6].

1.1 Statement of Problem

One serious problem in the network security is the managing of the firewall policy rules, the poor managing cause's conflicts of these rules. This study tackles four types of these conflicts dealing with anomalous policy rules in small network.

1.2 Objectives of Thesis

In this section, we present main objective and specific objectives of the research work.

1.2.1 Main Objective of Thesis

The main objective of this research is to develop a (SADAR) model for detection and filtering the anomalous of the firewall policy rules in small network in order to increases the accuracy factor in detecting these anomalous.

1.2.2 Specific Objectives of Thesis

1. To conduct in depth study of some papers and scientific research concerning the anomalous policy rules, and to review the most updated literature related.
2. Using linear search algorithm in (SADAR) model to detection the anomalous firewall policy rules.
3. To design a (SADAR) model that is able to detect the anomalous firewall policy rules for small networks.
4. To design and implement user interface suitable for the entry of new rules with constraints and verifications for the validity of structure.
5. To create manually dataset including four kinds from anomalous، such as (Shadowing Anomalous, Generalization Anomalous, Correlation Anomalous, and Redundancy Anomalous) and normal policy rules for testing and experiencing runs.
6. Evaluation, evaluate the experiencing runs in different situations with new policy rules, The accuracy factor in detecting anomalous firewall policy rules are evaluate.

1.3 Importance of the Thesis

1. Strengthen the accuracy factor for detect the anomalous firewall policy rules.
2. Reduce the mismanagement through network administrator.
3. Minimize time and effort in finding and detection the anomalous policy rules.
4. More constrains and validation in writing and management firewall policy rules

5. Develop user-friendly interface.
6. Decrease the weakness in network security of detection the anomalous policy rules.
7. Check up the policy rules intermediately before its adoption.
8. Work and use in independent environment as web application.

1.4 Scope and Limitations

1. Experimental tests to be conduct in Linux firewall version only (IP-Tables).
2. Detect the anomalous firewall policy rules in small networks.
3. This work is limited to individual firewall but not distributed firewalls policy rules.
4. Use top to down approach in detecting the anomalous rules, matching last rule with old rules as descending mechanism.

1.5 Thesis Structure

This thesis consists of six chapters: Introduction, Theory Background, Related Works, Methodology and Implementation, Experiments and Evaluation and finally, Conclusions and Future work. The main chapters are listed as below:

CHAPTER 1: Introduction

This chapter provides a brief introduction for our thesis, which explains the underlying concepts of firewall policy rules and anomalous of firewall policy rules and it talks about the thesis problem, the research objectives, the research importance, the research scope and limitations, as well as the research methodology.

CHAPTER 2: Theory Background

This chapter provides a brief introduction and an overview about firewall policy rules and anomalous rules, its definition, essential characteristics and the types of anomalous rules and examples.

CHAPTER 3: Related Works

This chapter provides an overview of the problem and solution in the recent related works of this thesis, there has been a significant amount of research in recent years to detect anomalous firewall policy rules, researchers apply different tools and approaches in detection anomalous process based.

CHAPTER 4: Methodology and Implementation

In this chapter, we presented our research methodology and built the (SADAR) model to strengthening accuracy in detection the anomalous rules. In implementation section, we describe the (SADAR) model, which we built, programing language, main libraries used in detection process.

CHAPTER 5: Experiments and Evaluation

In this chapter, we present details about the sets of experiments, and evaluate the experimental results. In addition, discussion for each set experiments and evaluation.

CHAPTER 6: Conclusions and Future work

In this chapter, we present the conclusion summarize the research achievement of experiments, and suggests future work

CHAPTER 2: Theory Background

This chapter provides a brief introduction and an overview about firewall policy rules and anomalous rules, its definition, essential characteristics and the types of anomalous rules and examples.

2.1 Overview of Firewall Policy Rule

In [22] define the firewall as a network security system between internal network and external network. As shown in Figure 2.1, either hardware or software based. That controls incoming and outgoing network traffic based on a set of policy rules and the firewall is be the first line of defense in network security. Many businesses and organizations protect their internal networks using firewalls and there is a several types of firewall technologies but our research depend on firewall Packet Filter technology.

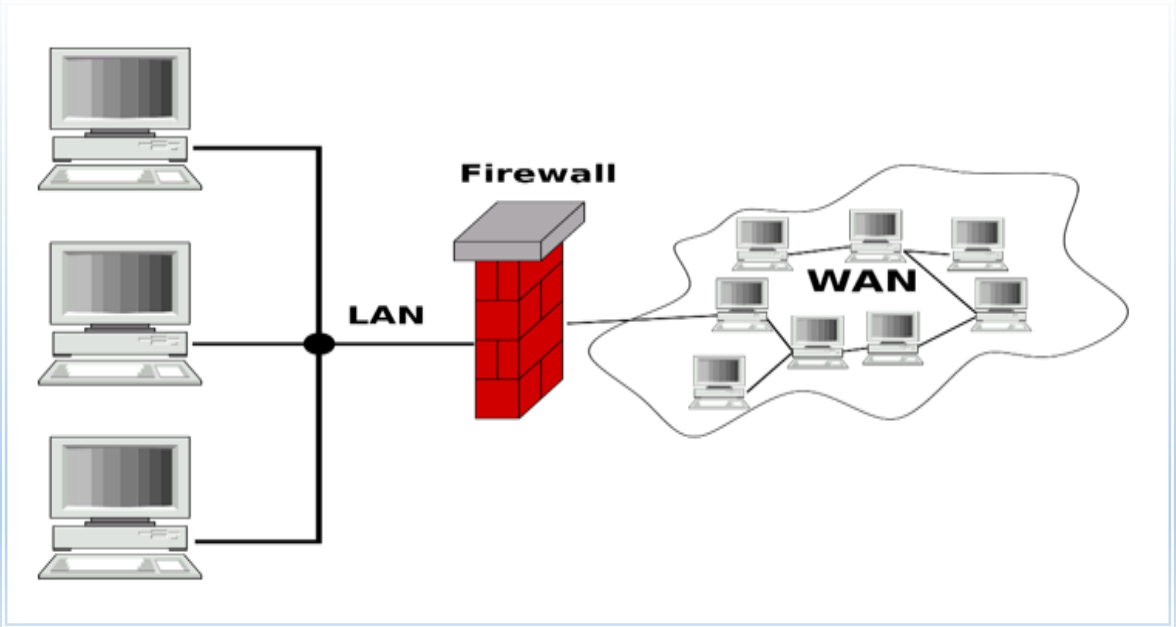

Figure 2.1: A Firewall between Internal Network and External Network [5]

2.2 The Mechanism of Firewall:

The firewall policy rule in the controlling a traversal of packets across the boundaries of a secured network, based on a specific security policy rule, a firewall policy rule is a list of ordered filtering rules that define the actions performed on matching packets over the action deny or accept the packet. The rules are composed of filtering fields (also called network fields) such as protocol type, source IP address, destination IP address, source port and destination port, and a filter action field. [22].

On other hand in [6] the filtering actions are either to accept, which allows the packet to be pass into or from the secure network, or to deny, which causes the packet to be discard, the packet is accept or denied by a specific rule, if the packet header information matches all the network fields of this rule. Otherwise, the next following rule is use to test the matching with this packet again. Similarly, this process repeated until a matching rule is a founded or the default policy action is a performed.

In [23] defined the packet filtering firewalls is operate at layer 3 of the OSI model. The packet filter firewall it receives packets and matching them with a set of list policy rules, the packet filtering firewalls provide more of protection for a network with reduce complications, the filtering packet is the process of passing or deny packets at a network interface based on source and destination addresses, ports, or protocols rather for action deny or accept, the new generation of packet filter can filter traffic based on many packet attributes as the following (source IP address, source port, destination IP address or port destination and service like www or FTP they can filter based on protocols).

In Figure 2.2, we discuss about the basic flow chart of packet filtering firewall of process income packet from external network to internal network, the packet filtering firewall receive the all packets come from external networks, the second step match every packet in individually process step by step with a set of list policy rules based some attributes as (source IP address, source port, destination IP address, port destination and protocol) based the action (deny or accept), if the packet match with rules and the action was be accept the firewall forwarding the packet to the destination, else if the action is denied then drop the packet, finally if the packet does not match with any rule of the list of rules the firewall is drop the packet[23].

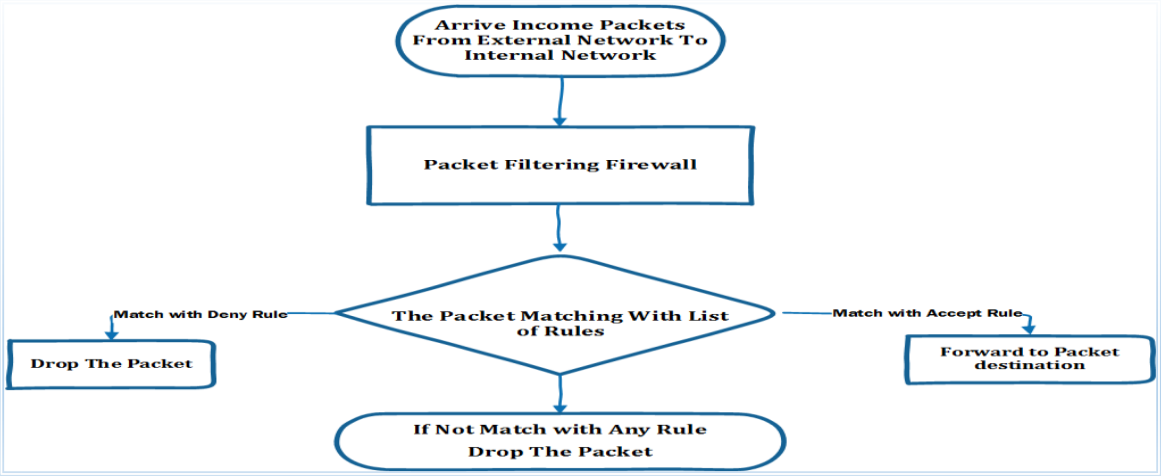

Figure 2.2: Basic Flow Chart of Income Packet Filtering Firewall [24]

In Figure 2.3, we discuss about the basic flow chart of packet filtering firewall of process outcome packet from internal network to external network, the packet filtering firewall receive the all packets arrive from internal networks, the second step match every packet in individually process step by step with a set of list policy rules based some attributes as (source IP address, source port, destination IP address, port destination and protocol) based the action (deny or accept), if the packet match with rules and the action was be accept the firewall forwarding the packet to the destination, else if the action is deny then drop the packet, finally if the packet not match with any rule of the list of rules the firewall is drop the packet[23].

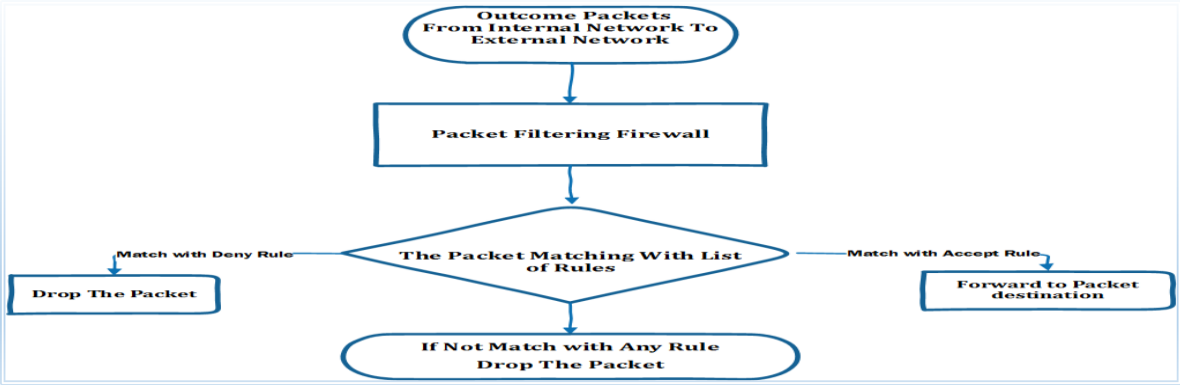

Figure 2.3: Basic Flow Chart of Outgoing Packet Filtering Firewall [24]

2.3 The Concepts of Policy Rules.

In [7] is define the rule as a set of criteria and an action to perform when a packet matches the criteria of a rule consist of the elements direction, protocol, source IP, source port, destination IP and destination port. Therefore, a complete rule may be define by the ordered direction, protocol, source IP, source port, destination IP, destination port and action. As shown in Figure 2.4 an example for rule as a set of criteria

	Source		Destination		
Protocol	Address	Port	Address	Port	Action
1: tcp,	140.192.37.20,	any,	*.*.*.*,	80,	deny
2: tcp,	140.192.37.*,	any,	*.*.*.*,	80,	accept
3: tcp,	*.*.*.*,	any,	161.120.33.40,	80,	accept
4: tcp,	140.192.37.*,	any,	161.120.33.40,	80,	deny
5: tcp,	140.192.37.30,	any,	*.*.*.*,	21,	deny
6: tcp,	140.192.37.*,	any,	*.*.*.*,	21,	accept
7: tcp,	140.192.37.*,	any,	161.120.33.40,	21,	accept
8: tcp,	*.*.*.*,	any,	*.*.*.*,	any,	deny
9: udp,	140.192.37.*,	any,	161.120.33.40,	53,	accept
10: udp,	*.*.*.*,	any,	161.120.33.40,	53,	accept
11: udp,	140.192.38.*,	any,	161.120.35.*,	any,	accept
12: udp,	*.*.*.*,	any,	*.*.*.*,	any,	deny

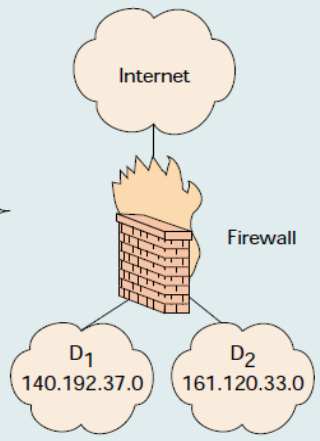

Figure 2.4: An Example For a Set Of Criteria As Rule [7]

2.4 Issus of Firewall Policy Rules

The size of the rules set varies according to the type and size of the organization, generally the rule set is very large because different network administrators often modify the policy rules according to their requirements.

These changes could cause the occurrence of anomalous, because of the large size of the rule set, it is difficult to detect anomalous by manually checking the rules one by one [17].

2.5 Formalization of Firewall Rule Relations.

In [8] Al-Shaer et al. presented and defined all the possible relations that may exit relate between two or more rules, the relations existing based on comparing the element of the same field between two different rules may be equal, inclusive or distinct.

Two values are equal if they exactly match, and are inclusive if one value is a subset of and not equal, the other (superset) and are distinct otherwise, two fields match if they are equal or inclusive, the diagrams shown in Figure 2.5 illustrates these relations between two filtering rules, Rx and Ry.

For examples, a source address value of 140.192.37.10 matches 140.192.37.∗ and does not match 140.192.37.20.

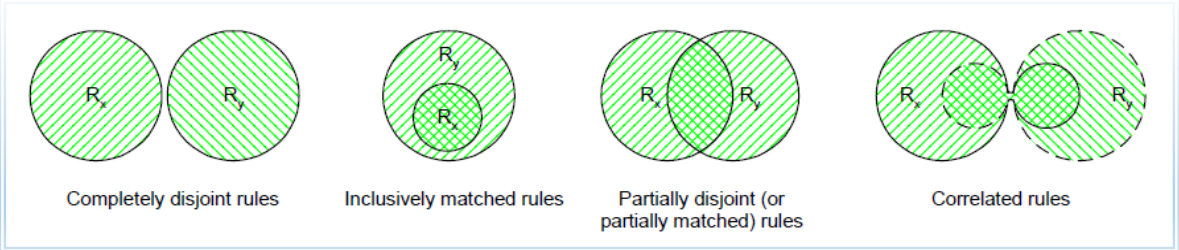

Figure 2.5: Relations between Two Filtering Rules Rx and Ry [8].

Definition 1 [8]: Rules Rx and Ry are exactly matched if every field in Rx is equal to the corresponding field in Ry. Formally: Rx exactly matches Ry if ∀ i: Rx[i] = Ry[i] where i ∈ {protocol, src_ip, src_port, dst_ip, dst_port}

For example, rule 1 and rule 2 below are exactly matched since all corresponding fields in both rules are equal.

1: tcp, 140.192.37.10, any, 163.122.51.∗, 21, accept

2: tcp, 140.192.37.10, any, 163.122.51.∗, 21, deny

Definition 2 [8]: Rules Rx and Ry are inclusively matched if they do not exactly match and if every field in Rx is a subset or equal to the corresponding field in Ry, In this relation, Rx is called the subset match while Ry is called the superset match.

Formally: Rx inclusively matches Ry if $\forall$ i : Rx[i] $\subseteq$ Ry[i] and $\exists$ j such that: Rx[j] $\neq$ Ry[j] where i, j $\in$ {protocol, src_ip, src_port, dst_ip, dst_port}

For example, rule 1 and rule 2 below are inclusively matched since they do not exactly match and every field in rule 1 is a subset or equal to the corresponding field in rule 2. Rule 1 is the subset match of the relation while rule 2 is the superset match.

1: tcp, 140.192.37.10, any, 163.122.51.*, 80, accept
2: tcp, 140.192.37.*, any, 163.122.51.*, any, deny

Definition 3 [8]: Rules Rx and Ry are completely disjoint if every field in Rx is not a subset and not a superset and not equal to the corresponding field in Ry.

Formally: Rx and Ry are completely disjoint if $\forall$ i: Rx[i] Ry[i] where $\in$ {$\subset$, $\supset$, =}, i $\in$ {protocol, src_ip, src_port, dst_ip, dst_port}

For example, rule 1 and rule 2 below are completely disjoint since all corresponding fields in both rules are distinct.

1: tcp, 140.192.37.10, 2000, 163.122.51.50, 80, accept
2: udp, 140.192.37.20, 3000, 163.122.51.60, 21, accept

Definition 4 [8]: Rules R_x and R_y are partially disjoint (or partially matched) if there is at least one field in R_x that is a subset or a superset or equal to the corresponding field in R_y, and there is at least one field in R_x that is not a subset and not a superset and not equal to the corresponding field in R_y.

Formally: R_x and R_y are partially disjoint (or partially matched) if $\exists$ i, j such that: R_x[i] R_y[i] and R_x[j] R_y[j] where $\in$ {$\subset$, $\supset$, =} and i, j $\in$ {protocol, src_ip, src_port, dst_ip, dst_port}

For example, rule 1 and rule 2 below are partially disjoint (or partially matched) since all fields in rule 1 are related to the corresponding fields in rule 2 except the destination port field.

1: tcp, 140.192.37.10, any, ∗.∗.∗.∗, 80, accept

2: tcp, 140.192.37.∗, any, ∗.∗.∗.∗, 21, deny

Definition 5 [8]: Rules Rx and Ry are correlated if some fields in Rx are subsets or equal to the corresponding fields in Ry, and the rest of the fields in Rx are supersets of the corresponding fields in Ry.

Formally: Rx and Ry are correlated if ∀ i: Rx[i] Ry[i] and ∃ i, j such that: Rx[i] ⊂ Ry[i] and Rx[j] ⊃ Ry[j] where ∈ {⊂, ⊃, =} and i, j ∈ {protocol, src_ip, src_port, dst_ip, dst_port}

For example, Rule 1 and rule 2 below are correlated since they have the same protocol, source and destination ports, and the source address of rule 1 is a subset of the corresponding fields in rule 2 , and the destination address of rule 1 is a superset of that of rule 2.

1: tcp, 140.192.37.10, any, ∗.∗.∗.∗, 80, accept

2: tcp, ∗.∗.∗.∗, any, 140.192.37.∗, 80, deny

2.6 Classifications of Anomalous Firewall Policy Rules

Alshaer and Hamed [8, 9 and 13] propose a classification of Anomalous in firewall policy rules as the following:

2.6.1 Shadowing Anomalies

One rule is have shadowing anomalies by another rules, whenever the rule which comes first in rule set matches all the packets and the second rule which is positioned after the first rule in rule set does not get chance to match any packet because the previous rule has matched all the packets.

It is a very important problem since the rule coming later to the previous rule will never be activate, important not when the traffic to be blocked will be allow or the traffic to be permit can be block.

2.6.2 Generalization Anomalies

Two rules that are in order one of them is have generalization of another if the first rules matches all the packets that can be also match by the second rule but the action performed is different in both of the rules.

If the order is reverse, the corresponding action will also be change. The rule, which comes later in the rule list, is Shadow by the previous rule and it has no effect on incoming packets. The super set rule is call General rule and the subset rule is call Specific rule.

If such generalization relation exists between two rules, then the superset rule should be place after the subset rule in the rule list.

2.6.3 Correlation Anomalies

One rule is have correlation anomalies by another rules, if both of them matches some common packets, the first rule matches some packets, which are also match by the second rule, the problem in this case is that the action performed by both of the rules is different.

It is a very important problem since then, in order to get the proper action, such correlated rules must be detect and should be specify with proper action to be perform.

2.6.4 Redundancy Anomalies

One rule is have redundant by another rules, if each one of them matches some packets and the also the same action, therefore, there is no effect on the firewall policy if one of redundant rules would be remove from the rule set.

It is very important to search and remove the redundant rules from the rule set because they increase the search time, space required to store the rule set and thus decrease the performance of the firewall, the firewall administrator should detect and remove such redundant rules to increase the performance of the firewall.

2.6.5 Irrelevance Anomalies

A filtering rule in a firewall is irrelevant if this rule cannot match any packets of traffic that might flow through this firewall, on other hand, the path between the source and destination addresses of this rule does not pass through the firewall and this rule has no effect on the filtering outgoing and income packet of this firewall.

2.7 An Examples of the Anomalous Firewall Policy Rules

In this section, we present examples about the anomalous firewall policy rules [14].

2.7.1 Example of Shadowing Anomalies

One or a set of previous rules that match all the packets, which also match the shadowed rule, while they perform a different action, can shadow a rule, in this case, all the packets that one rule intends to deny (accept) can be accepted (denied) by previous rule(s), thus the shadowed rule will never be taken effect

For example in Table 2.1, R4 is shadow by R3 because R3 allows every TCP packet coming from any port of 10.1.1.0/24 to the port 25 of 192.168.1.0/24, which is supposed to be deny by R4.

Table 2.1: Example of Shadowing Anomalies

No.	Protocol	Source IP And Port		Destination IP And Port		Action
R3	TCP	10.1.0.0/16	All	192.168.0.0/16	25	ACCEPT
R4	TCP	10.1.1.0/24	All	192.168.1.0/24	25	DROP

2.7.2 Example of Generalization Anomalies

A rule is a generalization of one or a set of previous rules if a subset of the packets matched by this rule is also match by the preceding rules but taking a different action.

For example in Table 2.2, R5 is a generalization of R4, These two rules indicate that all the packets from 10.1.1.0/24 are allow; except TCP packets from 10.1.1.0/24 to the port, 25 of 192.168.1.0/24 it is worth to be noted that generalization might not be an error.

Table 2.2: Example of Generalization Anomalies

No.	Protocol	Source IP And Port		Destination IP And Port		Action
R4	TCP	10.1.1.0/24	All	192.168.1.0/24	25	DROP
R5	All	10.1.1.0/24	All	0.0.0.0/0	All	ACCEPT

2.7.3 Example of Correlation Anomalies

One rule is correlate with other rules, if a rule intersects with others, but have a different action, in this case the packets match by the intersection of those rules, may be permitted by one rule, but denied by others.

For example in Table 2.3, R4 correlates with R5, and all UDP packets coming from any port of 10.1.1.0/24 to the port 53 of 172.32.1.0/24 match the intersection of these rules. Since R2 is a preceding rule of R5, every packet within the intersection of these rules is deny by R2. However, if their positions are swap, the same packets will be allow.

Table 2.3: Example of Correlation Anomalies

No.	Protocol	Source IP And Port		Destination IP And Port		Action
R2	UDP	10.1.0.0/16	All	172.32.1.0/24	53	DROP
R5	All	10.1.1.0/24	All	0.0.0.0/0	All	ACCEPT

2.7.4 Example of Redundancy Anomalies

A rule is redundant if there is another same or more general rule available that has the same action.

For example in Table 2.4, R1 is redundant with respect to R2 in table 2.4, since all UDP packets coming from any port of 10.1.2.0/24 to the Port 53 of 172.32.1.0/24 matched with R1 can match R2 as well with the same action.

Table 2.4: Example of Redundancy Anomalies

No.	Protocol	Source IP And Port		Destination IP And Port		Action
R1	UDP	10.1.2.0/24	All	172.32.1.0/24	53	DROP
R2	UDP	10.1.0.0/16	All	172.32.1.0/24	53	DROP

2.7.5 Example of Irrelevance Anomalies

The irrelevant anomalies, if this rule cannot match any packets traffic that might flow through this firewall, in this case exists when both of the source address and the destination address fields of the rule do not match with any packets traffic.

Summary

We presented in this chapter, an overview of the anomalous firewall policy rules, its definition, essential characteristics, and we presented all types of the anomalous firewall policy rules and examples about it, and issue of firewall policy rules. We discussed the network security problems through the anomalous firewall policy rules is becoming a bottleneck in network security.

CHAPTER 3: Related Works

This chapter provides an overview of the problem and solution in the recent related works of this thesis, there has been a significant amount of research in recent years to detect anomalous firewall policy rules, researchers apply different tools and approaches in detection anomalous process based.

Therefore, we classified some the related works according to these categories**:**

A. Tools of detecting firewall anomalous
B. Techniques of detecting firewall anomalous
C. Data mining in detecting firewall anomalous

A. Tools of detecting firewall anomalous

Luan. Y. et al. [18] the author's proposed a FIREMAN toolkit for detection the anomalous firewall policy rules, FIREMAN can detect anomalies rules on among multiple rules. Through analyzing the relationships between one rule and the collections of packet spaces derived from all preceding rules. The FIREMAN also has limitations in detecting anomalous firewall policy rules the FIREMAN only examines all preceding rules but ignores all subsequent policy rules when performing Anomalous.

On the other hand, **Pedditi S., et al. [19]** proposed a new protocol called FIEP (Firewall Information Exchange Protocol) works with distributed firewall, the FIEP works while considering parent-child relationships in detection the anomalous rules. The authors presented a simulated the protocol in Java with a static parent-child relationship. However the FIEP have many limitation need to change hardware/ software of existing firewalls and considering economic barriers and the FIEP is not implemented in real time, the results is still of simulation stage.

The author's **Al-Shaer and Hamed [9]** proposed and presented tool called a Firewall Policy Advisor for detection the anomalous firewall policy rules. is one of the earliest tools for firewall analysis, it used Binary Decision Diagrams (BDDs), but have a limitation in detection the anomalous firewall policy rules, only have the ability to detect the similar anomalies of policy rules, on the other hand ignored the other anomalous types.

B. Techniques of Detecting Firewall Anomalous

The author's **Khummanee et al. [20]** proposed and presented a Novel Firewall Rule Management Policy called Single Domain Decision Firewall (SDD) to verification the anomalous policy rules. However, the (SDD) have a limitation on solution, was as detection the rules have action acceptance in filtering the packets and ignore all deny action and implemented in virtual environment.

On the other hand, **Anbarasan et al. [10]** presented an Anomalous management framework for firewalls to detection the anomalies rules, using a rule-based segmentation technique in detection process, using packet space segmentation approach as the algorithm in Figure [3.1]. This technique is generating packet space segments for a set of firewall rules R, works by a disjoint packet space segments with rule, every packet segment associated with rules overlap relation among those rules. This approach not useful for small network, and not work in real-dynamic environment and need more equipment and effort in detection process.

Algorithm 1: Segment Generation for Network Packet Space of a Set of Rule R: Partition(R)

```
   Input: A set of rules, R.
   Output: A set of packet space segments, S.
1  foreach r ∈ R do
2      s_r ⟵ PacketSpace(r);
3      foreach s ∈ S do
4          /* s_r is a subset of s*/
5          if s_r ⊂ s then
6              S.Append(s \ s_r);
7              s ⟵ s_r;
8              Break;
9          /* s_r is a superset of s*/
10         else if s_r ⊃ s then
11             s_r ⟵ s_r \ s;
12         /* s_r partially matches s*/
13         else if s_r ∩ s ≠ ∅ then
14             S.Append(s \ s_r);
15             s ⟵ s_r ∩ s;
16             s_r ⟵ s_r \ s;
17     S.Append(s_r);
18 return S;
```

Figure3.1: Segment Generation for Network Packet Space Algorithm [10]

On the other hand, **Jeffrey_A. [15]** proposed a model checking the anomalies rules used a Binary Decision Diagrams to analyse firewall policy management and it was experimentally found that, the algorithm Binary Decision Diagrams; The algorithm for management in firewall IPTables format, the limitation of this research is the algorithm is classified as NP-Complete but is still not implemented.

The Authors, **Hongxin Hu et al.** [**14**] presented and implemented a policy analysis tool called Firewall Anomalous Management Environment (FAME), which too uses binary decision diagrams to represent firewall rule sets. FAME introduces a grid representation (matrix based) of the firewall anomalous, which provides a better understanding of policy anomalous. The experimental results can resolve 92 percent of the firewall conflicts; The FAME is unable to collect firewall rules in real-time dynamic systems. Moreover, FAME currently cannot handle in small firewalls rules.

C. Data Mining of Detecting Firewall Anomalous

The Authors, **Golnabi K. et al. [16]** using data mining technique to explore large amount of datasets of rules in detection process about the anomalous firewall policy rules, using association rule and frequency-based techniques, in provide a tool to analyze the network traffic from firewall log data files and rules. The mining technique also has limitation is needed large amount of dataset policy rules and firewall log data "packets space" files to be useful in detection anomalous firewall policy rules so; this is not efficient of firewall policy rules in small network because not have large amount of policy rules .

Summary

Most prior related works mentioned in this chapter introduced some solutions, but there are some limitations in detection anomalous policy rules, in [18] ignores all subsequent policy rules when performing anomalous, in [19] the solution is still of simulation stage, in [20] detection the rules have action acceptance only.

So, in our research, we consider these issues as possible, and we presenting strengthen the accuracy factor in detection the anomalous firewall policy rules, Therefore, we will be careful in applied the implementation of the proposed model, the (SADAR) model is classified as tool in detecting the anomalous rules.

CHAPTER 4: Methodology and Implementation

In this chapter, we presented a research methodology in Section 4.1, to produce the main objective of our (SADAR) model, and then we described in details the implementation in Section 4.2.

4.1 Research Methodology

In this section, the proposed (SADAR) model methodology was presented and built The (SADAR) model for strengthening the accuracy factor is a main and important task within created our (SADAR) model for small network security through detection the anomalous firewall policy rules.

The accuracy factor in detection the anomalous rules is still a bottleneck and main issues in detection the anomalous rules. Therefore, we are being careful in designing our (SADAR) model to be more accurate. The methodology consists of three main phases as follows:

1. Research and review.
2. Data collection.
3. Process of Proposed (SADAR) Model.

4.1.1 Research and review

In this section, we was survey and review some of the recent researches in previous chapter closely related to the thesis statement of problem, unlike previous researches we are going to design the (SADAR) model. We will consider these cons in previous research to overcome them in our research, in the following points:

1. Detection the anomalous rules within action acceptance and deny
2. Detection all of the anomalous firewall policy rules and the anomalous in sub rules.
3. Implement the detection process of every anomaly type as individual process to increases the accuracy factor.
4. Design interface to view every type of the anomalous rules in a separate table, to be the results are more accurate of review.
5. Build constraints for entry any new policy rules for the firewall to reduce mismanagement.

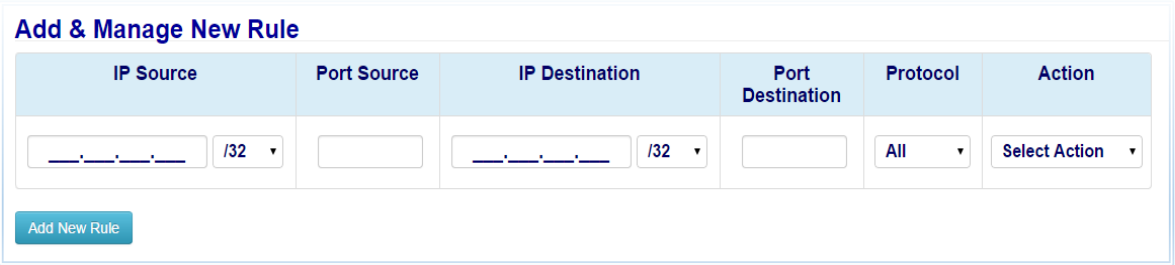

Figure 4.1: Add and Management New Rules

4.1.2 Data Collection

In this section, we built sixteen different dataset divided to four groups based on four different experiments according to anomalous types , all data sets are building based on six attributes of firewall policy rules, shown in Table 4.1 description of dataset attributes.

Table 4.1: Description of Dataset attributes [25]

Attributes Name	**Description**
IP Destination	IP Destination of the packet
IP Source	IP Source of the packet
Source port	Source port of the packet.
Destination port	Destination port of the packet
Protocol	Transport-layer protocol (i.e., TCP or UDP) of the packet
Action	Action to be performed on the traffic matching the rule (allow, deny)

On other hand, those groups is divided into four different size dataset and we suppose the size of dataset is more than 30 hosts and less than 200 hosts.

Every dataset is a created from different subnetworks; all datasets are divide into 30% anomalous rules and 70% normal rules, shown in Table 4.2 details of splitting the dataset.

Table 4.2: Details of Splitting the Dataset

Dataset No.	**Normal rules**	**Anomalous Rules**	**Total Rules**
1	140	60	200
2	84	36	120
3	42	18	60
4	21	9	30

4.1.3 Process of the Proposed (SADAR) Model

The process of the proposed (SADAR) A Model for **S**trengthening **A**ccuracy in **D**etection the **A**nomalous Firewall **R**ules in Small Network. is have three main section, to produces the main objectives of our research over on strengthening the accuracy factor in detecting the anomalous firewall policy rules in small network security, the main section of (SADAR) model are (Interaction Interface, Detection Process of (SADAR) Model And Repository) as below in Figure 4.2

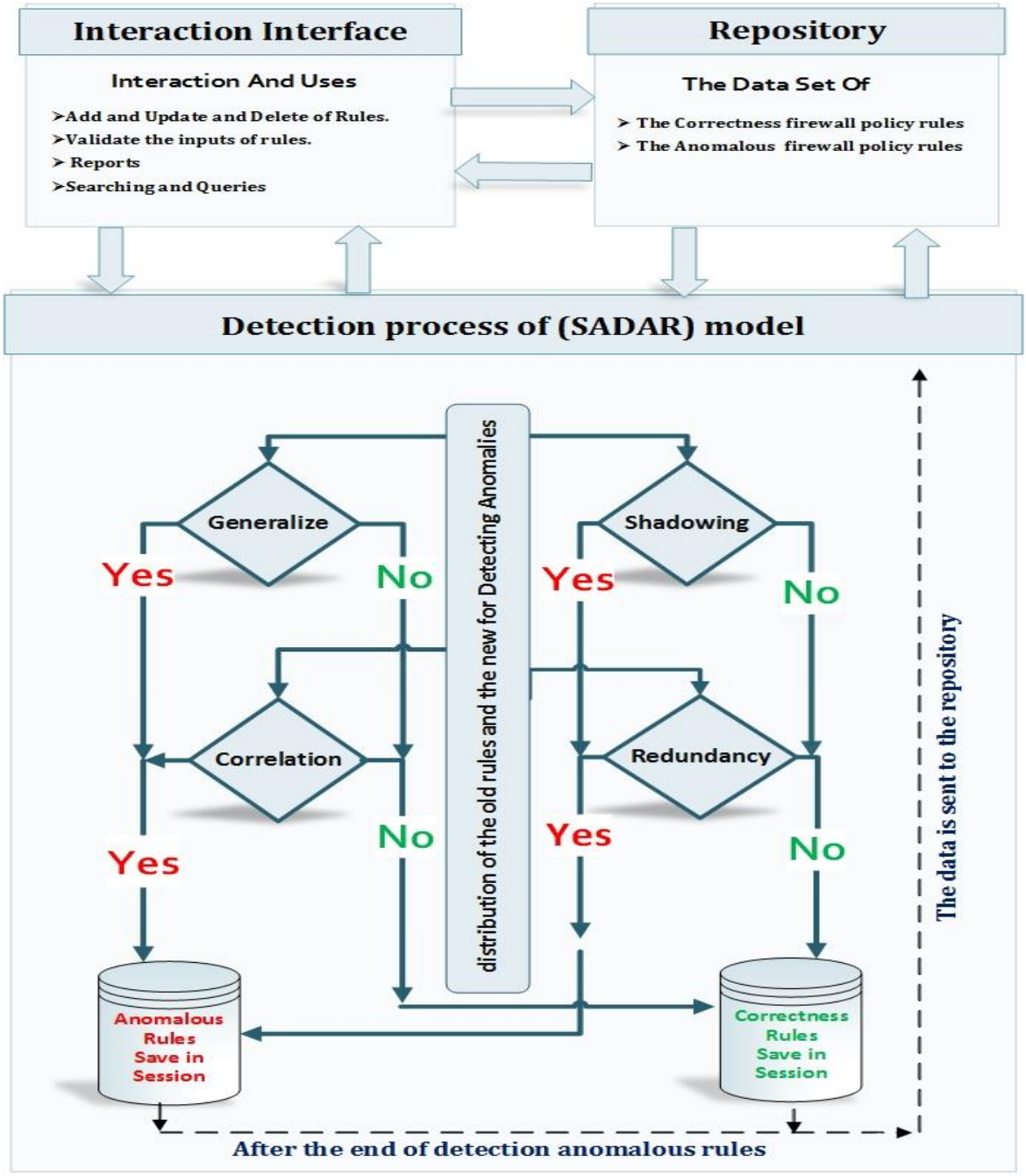

Figure 4.2: General View of Proposed (SADAR) Model

In this section, we present a description in the main sections of the process (SADAR) model, as below.

4.1.3.1 Interaction Interface Section

The first section of (SADAR) model have many function to help the administrator in managing and detection the anomalous firewall policy rules, thought the below points.

1. **Add, Update And Delete Of Rules**

   The administrators can managing the firewall policy rules thought interface easy to use, help the administrators in add any new rules and delete old rules and update any old policy rules.

2. **Validate the Inputs of Rules**

   Validate the inputs of the policy rules is necessary in managing the firewall to ensure the all input of rules is correctness data.

3. **Reports**

   Display reports about the anomalous rules.

4. **Searching and Queries**

   Help the administrators to reduce the time in the managing of the firewall policy rules we design searching part in the data set of the firewall policy rules.

4.1.3.2 Detection Process Section

The second section of the model have the main core processing of detection the anomalous rules. In this section have four part, every part customized for detection one type of anomalous.

These portions detection about four types of anomalous as the following: generalization anomaly, redundancy anomaly, shadowing anomaly and correlation anomaly. At the end of processing in detection the anomalous firewall policy rules save all anomalous in separate dataset to help the administrator in review the anomalous rules before add any new rules in future.

4.1.3.3 Repository

In repository section, there are two sub repositories as the following:

1. **Repository of Anomalous Rules.**

   There are link between the repository and interaction interface, in search query and present reports about the cases of anomalous rules, usefully of reviewing in reports over network administration, On other hand, after detection process, save detected, is saved in independent dataset, through the link between the detection process and repository.

2. **Repository of Correctness Rules**

   There are link between the repository and interaction interface, in search query and present reports about the correctness rules, usefully of reviewing in reports over network administration. On the other hand, after completion of detection process and validation of append the new rule does not conflict with any other rules is saved in independent dataset, to be as a list of rules, to use in matching between the packets and rules.

4.1.3.4 The Flow chart of (SADAR) Model

In this section, we discuss the main steps of functions for our "SADAR" model, through the below flow chart in Figure 4.3, as below.

1. Modify one or more new rules to check and detect the anomalous.
2. Submit query or search about old rules.
3. Validation all the input of new rules with stander rule format.
4. Share new rules to distribution process.
5. Share the current rules form dataset of firewall to distribution process.
6. Distribute and divide new and current rules to the four main parts to detect anomalous.
7. After detected all parts of anomalous detection process, if the new rule not have any anomalous with old rules store the new rules in dataset to preview and store.
8. Store all new rules into current rules dataset of firewall policy rules.
9. After finishing from all parts of anomalous detection, if the new rule have any anomalous with old rules store data set to preview.

10. Feedback from all new rules have anomalous to administrator.

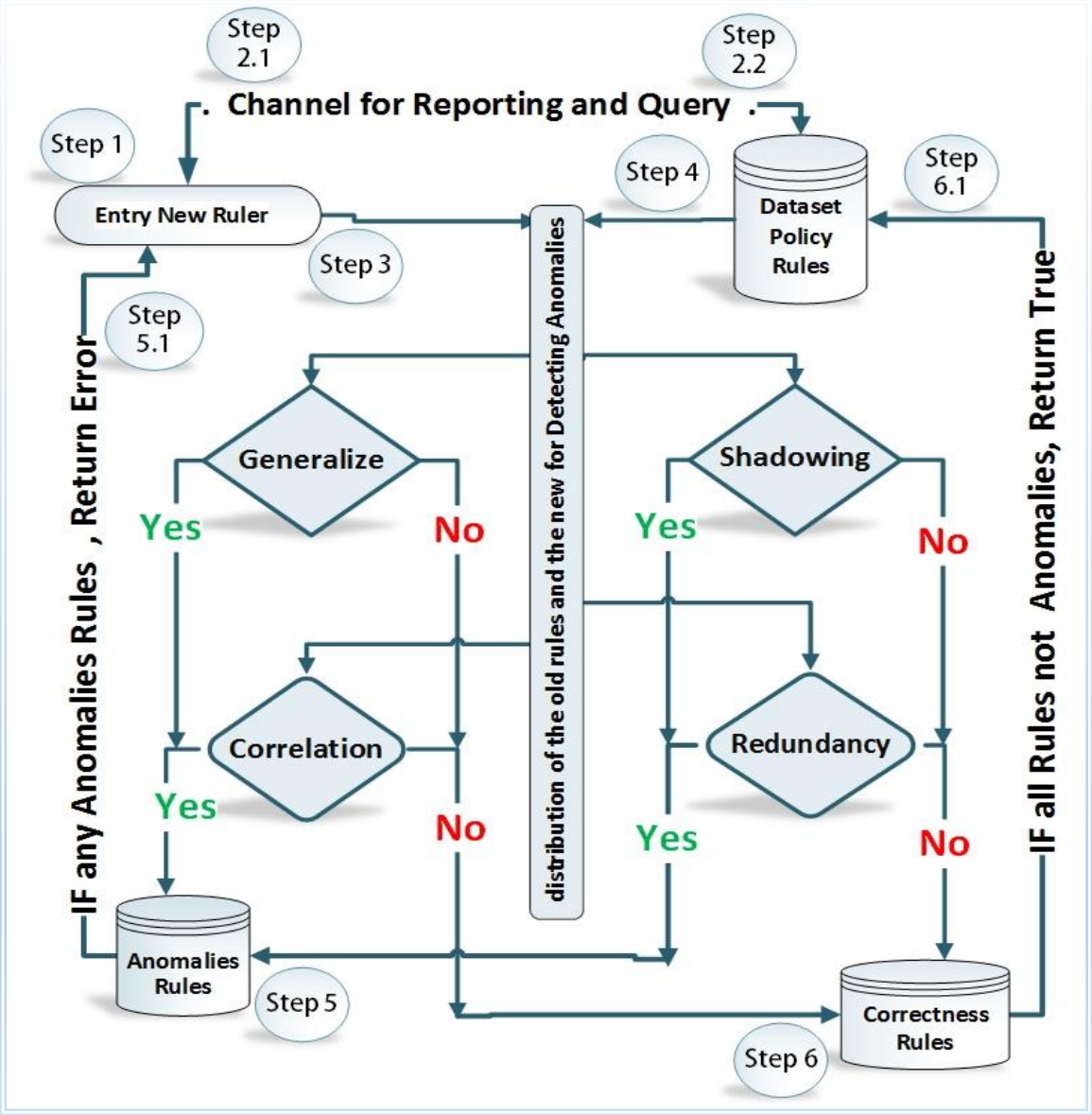

Figure 4.3: The Flow Chart of Proposed (SADAR) Model

4.2 Implementation of Proposed (SADAR) Model

In this section, we present the main points, is implementation of (SADAR) model, describe the programing language as used and explain how we used the linear search algorithm in detection the anomalous rules.

4.2.1 Programing language

We implement our (SADAR) Model on web application approach, because the web application is suitable can execution in any operating system, we used language programming PHP version 5.1.6 with Code-Igniter framework based MVC technique [27].

We programming and built four in dependent classes for detection four types of the anomalous policy rules, namely: (class to detection the generalization anomalous, class to detection the redundancy anomalous, class to detection the shadowing anomalous, class to detection the correlation anomalous)

4.2.2 Linear Search Algorithm

The Linear search algorithm or called as the sequential search algorithm is the basic search algorithm, used in data structures, linear search is used to find a particular element in an array or list of data, through matching between the goal element with other element [28].

We used the linear search algorithm in implement (SADAR) Model in detection the anomalous rules. We applied the algorithm in matching the new rule with old rules based on the attributes of rule as the following (Protocol, Source IP and Port, Destination IP and Port and Action) with all new rules and old rules.

In Table 4.3 an example of matching the attributes between new rules with old rules, we match every attribute of new rule with old rules in individual matching:

Table 4.3: An Example of Matching the Attributes between New Rules with Old Rule

New Rule	Old Rule
Action	Action
Protocol	Protocol
Destination Port	Destination Port
Source Port	Source Port
Source IP/Subnet Mask	Source IP/Subnet Mask
Destination IP/Subnet Mask	Destination IP/Subnet Mask

However, in matching new rule (source IP, destination IP) with old rule (source IP, destination IP), is still core issue, because is every anomaly type is have different conditions with other anomalous types, to apply the main goal of our research in strengthening accuracy factor in detect the anomalous rules.

Moreover, we use different mechanism in matching process, thought divided the IP address to four segments in array and matching every position segment with other position

segment in the same position and use the subnet mask to select the size of IP range, as the following in table 4.4.

Table 4.4: An Example for IP Address Segmentation and Matching

Rule No.	Source IP				Destination IP			
1	192	168	1	1	192	168	4	10
2	192	168	2	3	192	168	4	3

4.2.3 Pseudo Code Used in Implementation (SADAR) Model:

we present an overview of pseudo code for detection the anomalous firewall policy rules, over four independent classes used in implement our model, for detection four types of anomalous rules, as the below:

1. Pseudo code of detect the correlation anomalous.

The main core of this class is matching the overlapping between the new rules with old rules, based on if some fields of new rule are subsets or equal to the corresponding fields in old rule and some fields of new rule are supersets to the corresponding fields in old rule, and their actions are different**.**

2. Pseudo code of detect generalization anomalous

The main core of this class is matching the overlapping between the new rules with old rules, based on if every field of new rule is a superset or equal to the corresponding field in old rule and the actions are different.

3. Pseudo code of detect redundancy anomalous

The main core of this class is matching the overlapping between the new rules with old rules, based on if every field of new rule is a superset or equal to the corresponding field in old rule and both rules have the same action.

4. Pseudo code of detect shadowing anomalous

The main core of this class is matching the overlapping between the new rules with old rules, based on if every field of new rule is a superset to the corresponding field in old rule and their actions are different**.**

4.2.3.1 Pseudo Code of Detect the Correlation Anomalous

```
Class Correlation_Anomalous_Detection {

  Public function Correlation
($source_new_ip,$subnet_source_ip,$destination_new_ip,$subnet_destination_ip,

$source_new_port,$destination_new_port,$new_protocol,$new_action,$source_new_ip,
destination_new_ip) {

$source_new_ip = Segments ('.',$source_new_ip);

$destination_new_ip = Segments ('.',$destination_new_ip);


foreach (list-of-old not end){


$source_old_ip = Segments('.',$source_old_ip);

$destination_old_ip = Segments('.',$destination_old_ip);


IF ((Comparison ($old_protocol,$new_protocol) == 1) or ($new_protocol=='All') or ($old_protocol=='All'))
{

 IF ((Comparison ($old_action,$new_action) == 1)) {

 IF ((Comparison ($source_old_ip[0],$source_new_ip[0]) == 1) and

                    (Comparison ($source_old_ip[1],$source_new_ip[1]) == 1) and

                    (Comparison ($source_old_ip[2],$source_new_ip[2]) == 1) and

                    (Comparison ($source_old_ip[3],$source_new_ip[3]) == 1)) {

 IF ((Comparison ($destination_old_ip[0],$destination_new_ip[0]) == 1) and

                    (Comparison ($destination_old_ip[1],$destination_new_ip[1]) == 1) and

                    (Comparison ($destination_old_ip[2],$destination_new_ip[2]) == 1) and

                    (comparison ($destination_old_ip[3],$destination_new_ip[3]) == 1)) {


 IF (($source_old_port==$source_new_port)){


 IF (($destination_old_port==$destination_new_port)){


Correlation _Array[]=Old_rule;

//save the anomaly rule in array, then at the end export the array to private table in dataset

  }}}}}}}}}
```

4.2.3.2 Pseudo Code of Detect Generalization Anomalous

```
Class Generalization_Anomalous_Detection {

  Public function Generalization
($source_new_ip,$subnet_source_ip,$destination_new_ip,$subnet_destination_ip,

$source_new_port,$destination_new_port,$new_protocol,$new_action,$source_new_ip,
destination_new_ip) {

$source_new_ip = Segments ('.',$source_new_ip);

$destination_new_ip = Segments ('.',$destination_new_ip);

foreach (list-of-old not end){

$source_old_ip = Segments('.',$source_old_ip);

$destination_old_ip = Segments('.',$destination_old_ip);


IF ((Comparison ($old_protocol,$new_protocol) == 1) or ($new_protocol=='All') or ($old_protocol=='All'))
{


 IF ((Comparison ($old_action,$new_action) == 1)) {


 IF ((Comparison ($source_old_ip[0],$source_new_ip[0]) == 1) and
                    (Comparison ($source_old_ip[1],$source_new_ip[1]) == 1) and
                    (Comparison ($source_old_ip[2],$source_new_ip[2]) == 1) and
                    (Comparison ($source_old_ip[3],$source_new_ip[3]) == 1)) {


 IF ((Comparison ($destination_old_ip[0],$destination_new_ip[0]) == 1) and
                    (Comparison ($destination_old_ip[1],$destination_new_ip[1]) == 1) and
                    (Comparison ($destination_old_ip[2],$destination_new_ip[2]) == 1) and
                    (Comparison ($destination_old_ip[3],$destination_new_ip[3]) == 1)) {


IF (($source_old_port==$source_new_port)){


 IF (($destination_old_port==$destination_new_port)){


Generalization _Array[]=Old_rule;


//save the anomaly rule in array, then at the end export the array to private table in dataset

 }}}}}}}}}
```

4.2.3.3 Pseudo Code of Detect Redundancy Anomalous

```
Class Redundancy_Anomalous_Detection {

  Public function Redundancy
($source_new_ip,$subnet_source_ip,$destination_new_ip,$subnet_destination_ip,

$source_new_port,$destination_new_port,$new_protocol,$new_action,$source_new_ip,
destination_new_ip) {

$source_new_ip = Segments ('.',$source_new_ip);

$destination_new_ip = Segments ('.',$destination_new_ip);


foreach (list-of-old not end){


$source_old_ip = Segments('.',$source_old_ip);

$destination_old_ip = Segments('.',$destination_old_ip);

IF ((Comparison ($old_protocol,$new_protocol) == 0) or ($new_protocol=='All') or ($old_protocol=='All'))
{


 IF ((Comparison ($old_action,$new_action) == 1)) {


 IF ((Comparison ($source_old_ip[0],$source_new_ip[0]) == 1) and
                    (Comparison ($source_old_ip[1],$source_new_ip[1]) ==1) and
                    (Comparison ($source_old_ip[2],$source_new_ip[2]) == 1) and
                    (Comparison ($source_old_ip[3],$source_new_ip[3]) == 1)) {


 IF ((Comparison ($destination_old_ip[0],$destination_new_ip[0]) == 1) and
                    (Comparison ($destination_old_ip[1],$destination_new_ip[1]) == 1) and
                    (Comparison ($destination_old_ip[2],$destination_new_ip[2]) == 1) and
                    (Comparison ($destination_old_ip[3],$destination_new_ip[3]) == 1)) {


 IF (($source_old_port==$source_new_port)){


 IF (($destination_old_port==$destination_new_port)){


Redundancy _Array[]=Old_rule;

//save the anomaly rule in array, then at the end export the array to private table in dataset

 }}}}}}}}}
```

4.2.3.4 Pseudo Code of Detect Shadowing Anomalous

```
Class Shadowing_Anomalous_Detection {

  Public function Shadowing
($source_new_ip,$subnet_source_ip,$destination_new_ip,$subnet_destination_ip,

$source_new_port,$destination_new_port,$new_protocol,$new_action,$source_new_ip,
destination_new_ip) {

$source_new_ip = Segments ('.',$source_new_ip);

$destination_new_ip = Segments ('.',$destination_new_ip);


foreach (list-of-old not end){


$source_old_ip = Segments('.',$source_old_ip);

$destination_old_ip = Segments('.',$destination_old_ip);

IF ((Comparison ($old_protocol,$new_protocol) == 1) or ($new_protocol=='All') or ($old_protocol=='All'))
{


 IF ((Comparison ($old_action,$new_action) == 1)) {


 IF ((Comparison ($source_old_ip[0],$source_new_ip[0]) == 1) and
                    (Comparison ($source_old_ip[1],$source_new_ip[1]) == 1) and
                    (Comparison ($source_old_ip[2],$source_new_ip[2]) == 1) and
                    (Comparison ($source_old_ip[3],$source_new_ip[3]) == 1)) {


 IF ((Comparison ($destination_old_ip[0],$destination_new_ip[0]) == 1) and
                    (Comparison ($destination_old_ip[1],$destination_new_ip[1]) == 1) and
                    (Comparison ($destination_old_ip[2],$destination_new_ip[2]) == 1) and
                    (Comparison ($destination_old_ip[3],$destination_new_ip[3]) == 1)) {


 IF (($source_old_port==$source_new_port)){


 IF (($destination_old_port==$destination_new_port)){


Shadowing _Array[]=Old_rule;

//save the anomaly rule in array, then at the end export the array to private table in dataset

 }}}}}}}}}
```

Summary

In this chapter, we presented our research methodology and built the (SADAR) model to strengthening accuracy factor in detection the anomalous rules, and described the main section of model, in implementation section; we described the (SADAR) model, which we built, programing language, main classes used in detection process.

CHAPTER 5: Experiments and Evaluation.

In this chapter, we presented our experiments, then evaluation the accuracy factor of (SADAR) model, to ensure the main objective of our research through the strengthening the accuracy factor in detecting the anomalous firewall policy rules, and detect any anomalous rules modification or updating

5.1 Experiments Setup

In this section, we present an overview about the dataset, experimental environment and tools used in (SADAR) model experiment, as the following:

5.1.1 Dataset collection.

The (SADAR) model is applied on different size of dataset over anomalous rules, we collected a similar dataset used in previous research in chapter 3, contain four types of anomalous rules, as shadowing, generalization, correlation and redundancy anomalous rules and correct rules, show in Table 5.1 sample dataset of firewall policy rules.

Table 5.1: Sample Dataset of Firewall Policy Rules.

No.	Protocol	Source IP & Port		Destination IP & Port		Action
R1	UDP	10.1.2.0/24	All	172.32.1.0/24	53	DROP
R2	UDP	10.1.0.0/16	All	172.32.1.0/24	53	DROP
R3	TCP	10.1.0.0/24	All	192.168.0.0/16	25	ACCEPT
R4	TCP	10.1.0.0/24	All	192.168.0.0/16	25	ACCEPT
R5	UDP	10.1.0.0/24	All	192.168.0.0/16	25	DROP
R6	TCP	10.1.0.0/24	All	192.168.0.0/16	25	ACCEPT
R7	TCP	192.168.0.0/16	All	192.168.1.0/24	25	DROP
R8	All	10.1.1.0/24	All	0.0.0.0/0	All	ACCEPT

5.1.2 Experimental Environment and Tools.

Our experiments are conducted with a laptop PC, running windows 8.1 x64 operating system with core i5- 3230M CPU, @ 2.60 GHz, with 8 GB memory RAM.

In (SADAR) model experiments being conducted using the following equipment and software.

5.1.2.1 Design and Build Real Small Network

We design and bullied real small network environment, this network have central firewall policy rules for control the traversal the packets in external and internal network.

The network have many sub networks to use different levels of IP address network to be more complicated in the managing, this leads us to testing larger numbers of cases of anomalous and check the accuracy in detecting the anomalous, as shown in Figure 5.1.

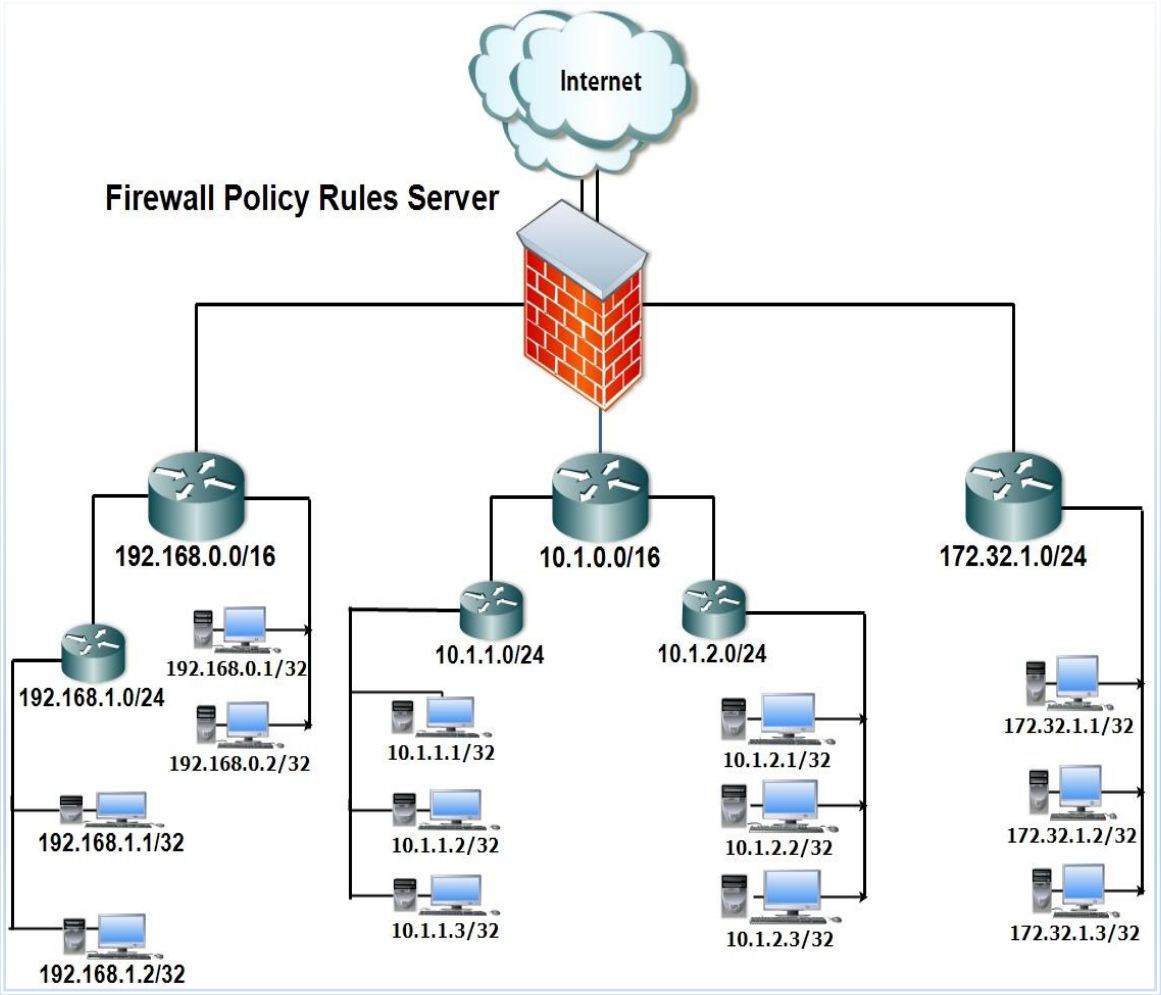

Figure 5.1: The Main Structure of Small Network Used in Our Experiment

5.1.2.2 Operating System and Firewall

We used Linux operating system and use IPTables last version of firewall under Linux operating to apply the rules and we use apache server.

5.1.2.3 Programming Languages and Software

We use php programing language with Code-Igniter framework and MySQL database application and we use Firefox and Google Chrome web browser in execution (SADAR) model.

5.2 Experiment and Results of (SADAR) Model.

In this section, we present sixteen different experiment, those experiments are divided into four different groups of experiments according to four types of anomalous rules, and the details of these experiments are explained as follows:

5.2.1 Experiment and Result of Shadowing Anomalies

In this section, we applied four different experiments according into different dataset of the firewall policy rule, the dataset of firewall policy rules is building with different cases according to the overlapping of the anomalous of shadowing type and the dataset is divided into 70% normal rules and 30% anomalous rules. Show in Table 5.2 illustrates the results of four experiments according into the detection the shadowing anomalous rules.

Table 5.2: Experiments Results on Detection the Shadowing Anomalous Rules

Experiment No.	Dataset Description Of Firewall Policy Rules				Experiments Results	
	Dataset No.	Normal Rules	Anomalous Rules	Total Rules	Detected Rules	Time (MS)
1	1	140	60	200	57	0.6460
2	2	84	36	120	33	0.1996
3	3	42	18	60	16	0.0454
4	4	21	9	30	7	0.0120

On the other hand, in these experiments we applied the overlapping of shadowing anomalies, if one or a set of previous rules that match all the packets while they perform a different action, in this case, all the packets that one rule intends to (deny/accept) can be accepted (denied) by previous rules, the shadowed rule will never be taken effect.

Shown in Table 5.3, sample of shadowing anomalous rules, for example, the R2 is shadow by R1 because R1 allows every TCP packet coming from any port of 10.1.1.0/24 to the port 25 of 192.168.1.0/24, which is supposed to be deny by R2.

Table 5.3: Sample Dataset of shadowing anomalous rules

No.	Protocol	Source IP And Port		Destination IP And Port		Action
R1	TCP	10.1.0.0/16	All	192.168.0.0/16	25	ACCEPT
R2	TCP	10.1.1.0/24	All	192.168.1.0/24	25	DROP

5.2.2 Experiment and Result of Generalization anomalies

In this section, we applied four different experiments according into different dataset of the firewall policy rule, the dataset of firewall policy rules is building with different cases according to the overlapping of the anomalous of generalization type and the dataset is divided into 70% normal rules and 30% anomalous rules. Show in Table 5.4 illustrates the results of four experiments according into the detection the generalization anomalous rules.

Table 5.4: Experiments Results on Detection the Generalization Anomalous Rules

Experiment No.	Dataset Description Of Firewall Policy Rules				Experiments Results	
	Dataset No.	Normal Rules	Anomalous Rules	Total Rules	Detected Rules	Time (MS)
1	1	140	60	200	57	0.3912
2	2	84	36	120	33	0.1434
3	3	42	18	60	16	0.0369
4	4	21	9	30	8	0.0101

On the other hand, in these experiments, we applied the overlapping of generalization anomalies, if one or a set of previous rules and a subset of the packets matched by this rule is also matched by the previous rules but taking a different action.

shown in Table 5.5, sample dataset of generalization anomalies , For Example, R2 is a generalization of R1, These two rules indicate that all the packets from 10.1.1.0/24 are

allow; except TCP packets from 10.1.1.0/24 to the port, 25 of 192.168.1.0/24 it is worth to be noted that generalization might not be an error.

Table 5.5: Sample Dataset of Generalization Anomalous Rules

No.	Protocol	Source IP And Port		Destination IP And Port		Action
R1	TCP	10.1.1.0/24	All	192.168.1.0/24	25	DROP
R2	All	10.1.1.0/24	All	0.0.0.0/0	All	ACCEPT

5.2.3 Experiment and Result of Correlation Anomalous

In this section, we applied four experiments according into different dataset of the firewall policy rule, the dataset of firewall policy rules is building with different cases according to the overlapping of the anomalous of correlation type and the dataset is divided into 70% normal rules and 30% anomalous rules. Show in Table 5.6 illustrates the results of four experiments according into the detection the correlation anomalous rules.

Table 5.6: Experiments Results of Detection the Correlation Anomalous Rules

Experiment No.	Dataset Description Of Firewall Policy Rules				Experiments Results	
	Firewall No.	Norma Rules	Anomalous Rules	Total Rules	Detected Rules	Time (MS)
1	1	140	60	200	55	0.3567
2	2	84	36	120	31	0.1318
3	3	42	18	60	15	0.0329
4	4	21	9	30	7	0.0103

On the other hand, in these experiments we applied the overlapping of correlation anomalies, if one or a set of rules is correlation with other rules, if a rule intersects with others but taking a different action, in this case the packets matched by the intersection of those rules may be allow by one rule, but denied by others.

Shown in Table 5.7, sample of correlation anomalous, For Example, R1 correlation with R2, and all UDP packets coming from any port of 10.1.1.0/24 to the port 53 of 172.32.1.0/24 match the intersection of these rules, since R1 is a previous rule of R2, every

packet within the intersection of these rules is deny by R1. However, if their positions are swap, the same packets will be allow.

Table 5.7: Sample Dataset of Correlation Anomalous Rules

No.	Protocol	Source IP And Port		Destination IP And Port		Action
R1	UDP	10.1.0.0/16	All	172.32.1.0/24	53	DROP
R2	All	10.1.1.0/24	All	0.0.0.0/0	All	ACCEPT

5.2.4 Experiment and Result of Redundancy Anomalies

In this section, we applied four experiments according into different dataset of the firewall policy rule, the dataset of firewall policy rules is building with different cases according to the overlapping of the anomalies of redundancy type To ensure all possible cases in this type and the dataset is divided into 70% normal rules and 30% anomalous rules. Show in Table 5.8 illustrates the results of four experiments according into the detection the redundancy anomalous rules.

Table 5.8: Experiments Results on Detection the Redundancy Anomalous Rules

Experiment No.	Dataset Description Of Firewall Policy Rules				Experiments Results	
	Dataset No.	Normal Rules	Anomalous Rules	Total Rules	Detected Rules	Time (MS)
1	1	140	60	200	58	0.6729
2	2	84	36	120	35	0.1631
3	3	42	18	60	17	0.0891
4	4	21	9	30	8	0.0562

On the other hand, in these experiments we applied the overlapping of redundancy anomalies, If there is another same or more general rule available, that has the taking (action).

Shown in Table 5.9, sample of redundancy anomalous rules, R1 is redundant with respect to R2, since all UDP packets coming from any port of 10.1.2.0/24 to the Port 53 of 172.32.1.0/24 matched with R1 can match R2 as well with the same action.

Table 5.9: Sample of Redundancy Anomalous Rules

No.	Protocol	Source IP And Port		Destination IP And Port		Action
R1	UDP	10.1.2.0/24	All	172.32.1.0/24	53	DROP
R2	UDP	10.1.0.0/16	All	172.32.1.0/24	53	DROP

5.3 The Evaluation the Proposed (SADAR) Model

In this section, performance evaluation of the (SARDAR) model in detection the anomalous firewall rules is one of the most important tasks in our research based the overall accuracy, for evaluating the experiments and results.

We use confusion matrix because it extracts the results and computing the accuracy, detection rate, classification error, and f-measure.

A confusion matrix shows the number of correct and incorrect predictions made by the classification model compared to the actual outcomes (target value) in the data [1], there are four estimates define the members of the matrix as the below:

True Positive (TP): refer to number of positive instances that correctly labeled by the model

False Positive (FP): refer to number of negative instances that were incorrectly labeled by the model.

True Negative (TN): refer to number of negative instances that correctly label by the model.

False Negative (FN): refer to number of positive instances that were incorrectly labeled by the model

Table 5.10: Confusion Matrix [4]

Confusion Matrix		Target	
		Positive	Negative
Model	Positive	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)

Also, the overall accuracy the most commonly to evaluate performance of our model in detection the anomalous firewall rules and other measures using confusion matrix to evaluate the performance which as following:

Detection Rate: refer to percentage of positive instances that correctly labeled by the model [4].

$$\text{Detection Rate} = \frac{No.\ Anomalies \times |TP| + No.\ Normal \times |TN|}{No.\ Anomalies + No.\ Normal} \quad \text{(Eq. 5.1)}$$

Classification Error: refer to relative number of misclassified, and the rate was recorded on the training and testing data sets [4].

$$\text{Classification Error} = \frac{FP+FN}{TP+TN} = 100 - overall\ accuracy \quad \text{(Eq. 5.2)}$$

Recall: refer to number of positive instances that correctly labeled by the model [4].

$$Recall = \frac{TP}{TP+FN} \quad \text{(Eq. 5.3)}$$

Precision: refer to the percentage of retrieved instances that are relevant [4].

$$Precision = \frac{TP}{TP+FP} \quad \text{(Eq. 5.4)}$$

Overall Accuracy: refer the percentage of test set tuples that are correctly classified by the model [4].

$$Overall\ Accuracy = \frac{|TP|+|TN|}{|TP|+|FP|+|TN|+|FN|} \quad \text{(Eq. 5.5)}$$

F-measure: refer to the harmonic mean of precision and recall [4].

$$F-measure = \frac{2}{\frac{1}{Precision}+\frac{1}{Recall}} = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad \text{(Eq. 5.6)}$$

True Positive Rate: refer to number of positive instances that correctly labeled by the model [4].

$$True\ Pasitive\ rate = \frac{TP}{TP+FN} \quad \text{(Eq. 5.7)}$$

False Positive Rate: refer to number of negative instances that were incorrectly labeled by the model [4].

$$False\ Positive\ rate = \frac{FP}{TN+FP} \quad \text{(Eq. 5.8)}$$

True Negative Rate: refer to number of negative instances that correctly label by the model [4].

$$True\ Negative\ rate = \frac{TN}{TN+FP} \quad \text{(Eq. 5.9)}$$

False Negative Rate: refer to number of positive instances that were incorrectly labeled by the model [4].

$$False\ Negative\ rate = \frac{FN}{TP+FN} \quad \text{(Eq. 5.10)}$$

5.3.1 Evaluation the Experiment and Result of Shadowing Anomalies

In this section, we present the performance evaluation four different experiments according to the detection of shadowing anomalous rules, we used the confusion matrix into measuring the performance evaluation the detection of shadowing anomalous rules in previous experiments results, Table 5.11 and Figure 5.2 illustrates the results of the performance evaluation (detection of shadowing anomalous rules).

Table 5.11: Results of the performance Evaluation of Shadowing Anomalies

Description	Exp. NO. 1	Exp. NO. 2	EXP. NO. 3	EXP. NO. 4	EXP. Average
Overall Accuracy	96.45	94.11	92.17	84.54	**91.82%**
Classification Error	3.55	5.89	7.83	15.46	**8.18%**
Detection Rate	95.87	93.13	90.86	81.84	**90.42%**
Recall	96.45	94.11	92.17	84.54	**91.82%**
Precision	95.00	91.67	88.89	77.78	**88.33%**
F-measure	95.72	92.87	90.50	81.02	**90.03%**

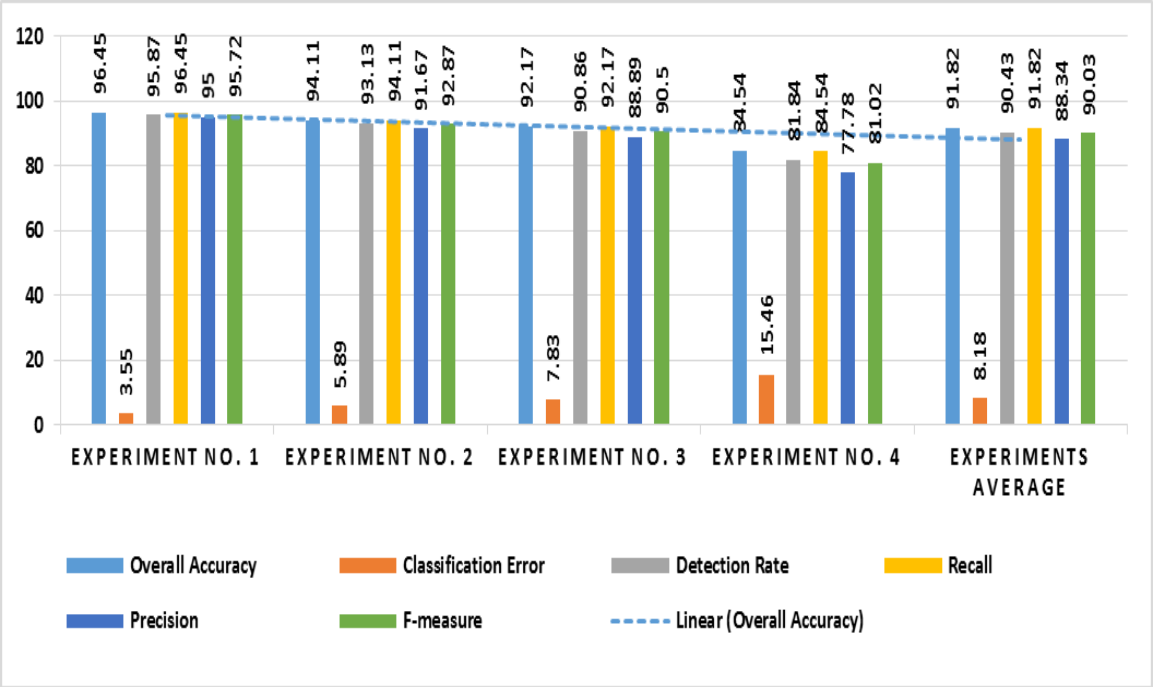

Figure 5.2: Results of the Performance Evaluation of Shadowing Anomalies

We note the result of the evaluation measuring is different in every experiment, these results are logical according to the composition of dataset used in previous experiments, these dataset generate based different collection from subnetworks and depend on the ordering of the rules can affect significantly in detection between the anomalous rules with normal rules.

On other hand, the results of evaluation measuring are affected based to the overlapping complexity between the anomalous and normal rules according to the anomalous types.

The shadowing anomalous rules have a medium degree of overlapping complexity between the rules and this affects in the results of evaluation measuring, comparing with another type of the anomalous rules, the overlapping complexity between the rules depends as the following: Each field in Ry is compare to the corresponding field in Rx and if every field of Rx is a superset or equal to the corresponding field in Ry and their actions are different, therefore; these factors have medium effect in the detection more anomalous rules.

Therefore, the shadowing anomalous rules have the average of overall accuracy is 91.82% and this is a medium rate comparing with another type of the anomalous rules.

5.3.2 Evaluation the Experiment and Result of Generalization Anomalies

In this section, we present the performance evaluation four different experiments according to the detection of generalization anomalous rules.

We used the confusion matrix into measuring the performance evaluation the detection of generalization anomalous rules in previous experiments results, Table 5.12 and Figure 5.3 illustrates the results of the performance evaluation (detection the generalization anomalous).

Table 5.12: Results of the Performance Evaluation of Generalization Anomalies

Description	EXP. NO. 1	EXP. NO. 2	EXP. NO. 3	EXP. NO. 4	Experiments average
Overall Accuracy	96.45	94.11	92.17	92.17	93.73
Classification Error	3.55	5.89	7.83	7.83	6.27
Detection Rate	95.87	93.13	90.86	90.86	92.68
Recall	96.45	94.11	92.17	92.17	93.73
Precision	95.00	91.67	88.89	88.89	91.11
F-measure	95.72	92.87	90.50	90.50	92.40

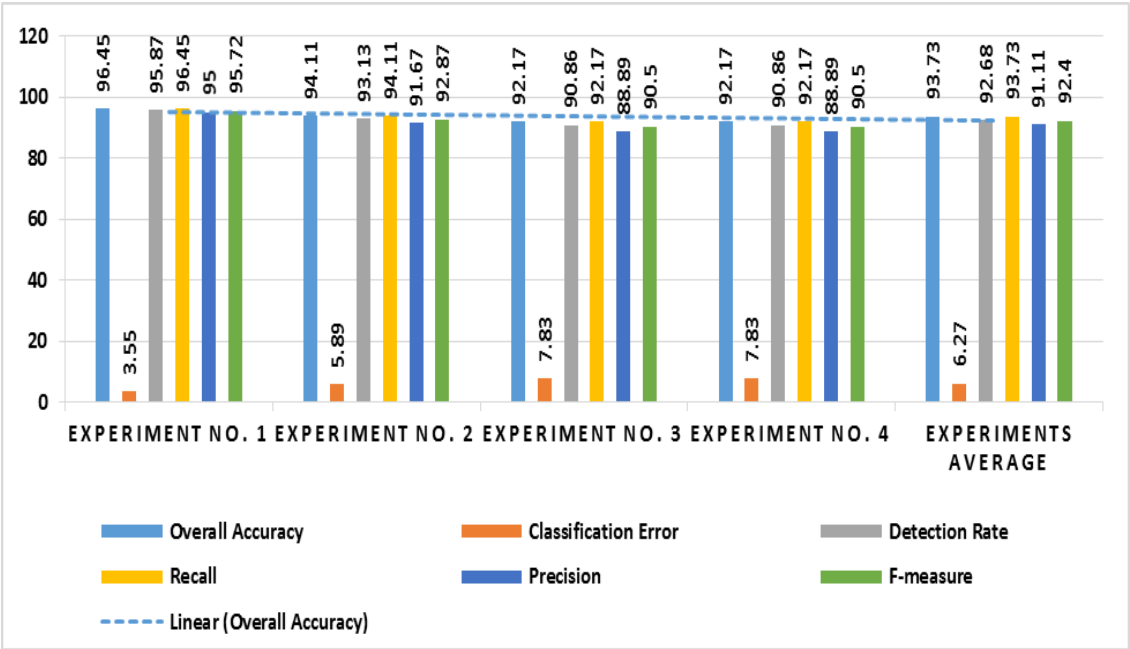

Figure 5.3: Results of the Performance Evaluation of Generalization Anomalies

We note the result of the evaluation measuring is different in every experiment, these results are logical according to some factors as:

- The composition of dataset used in previous experiments, these dataset generate based different collection from subnetworks.
- The ordering of the rules is affect significantly in detection between the anomalous rules with normal rules.
- The degree of overlapping complexity between the anomalous and normal rules can affects significantly in evaluation results.

On other hand, the generalization anomalous rules have a medium degree of overlapping complexity between the rules, comparing with another type of the anomalous rules; the overlapping complexity between the rules depends as the following:

If every field of Rx is a superset or equal to the corresponding field in Ry and the actions are different.

Therefore, the generalization anomalous rules have the average of overall accuracy is 93.73% and this is a medium rate comparing with another type of the anomalous rules.

5.3.3 Evaluation the Experiment and Result of Correlation Anomalies

In this section, we present the performance evaluation four different experiments according to the detection of correlation anomalous rules.

We used the confusion matrix into measuring the performance evaluation the detection of correlation anomalous rules in previous experiments results, Table 5.13 and Figure 5.4 illustrates the results of the performance evaluation (detection of correlation anomalous rules).

Table 5.13: Results of the Performance Evaluation of Correlation Anomalies

Description	EXP. NO. 1	EXP. NO. 2	EXP. NO. 3	EXP. NO. 4	Experiments Average
Overall Accuracy	94.11	90.25	88.33	84.54	**89.31**
Classification Error	5.89	9.75	11.67	15.46	**10.69**
Detection Rate	93.13	88.59	86.33	81.84	**87.47**

Recall	94.11	90.25	88.33	84.54	**89.31**
Precision	91.67	86.11	83.33	77.78	**84.72**
F-measure	92.34	88.13	85.76	81.02	**86.81**

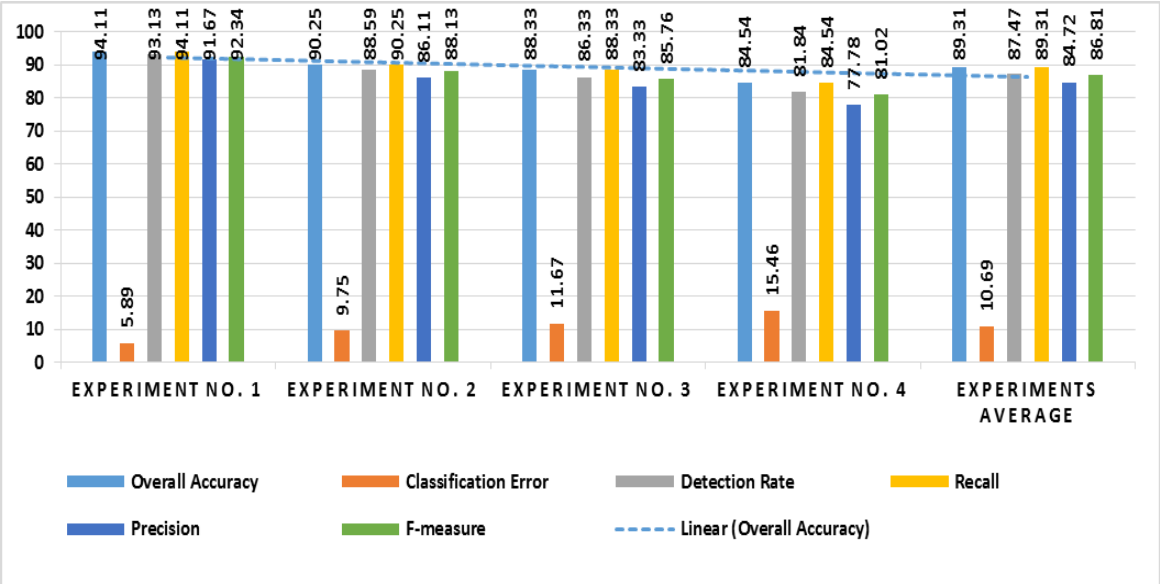

Figure 5.4: Results of the Performance Evaluation of Correlation Anomalies

We note the result of the evaluation measuring is different in every experiment, these results are logical according to some factors as:

- The composition of dataset used in previous experiments, these dataset generate based different collection from subnetworks.
- The ordering of the rules is affect significantly in detection between the anomalous rules with normal rules.
- The degree of overlapping complexity between the anomalous and normal rules can affects significantly in evaluation results.

On other hand, the correlation anomalous rules have a high degree of overlapping complexity between the rules, comparing with another type of the anomalous rules; the overlapping complexity between the rules depends as the following:

If some fields of Rx are subsets or equal to the corresponding fields in Ry and some fields of Rx are supersets to the corresponding fields in Ry, and their actions are different, then Rx is correlation with Ry

Therefore, the correlation anomalous rules have the average of overall accuracy is 89.31% and this is a low average comparing with another type of the anomalous rules.

5.3.4 Evaluation the Experiment and Result of Redundancy Anomalies

In this section, we present the performance evaluation four different experiments according to the detection of redundancy anomalous rules, we used the confusion matrix into measuring the performance evaluation the detection of redundancy anomalous rules in previous experiments results, Table 5.14 and Figure 5.5 illustrates the results of the performance evaluation (detection of redundancy anomalous rules).

Table 5.14: Results of the Performance Evaluation of Redundancy Anomalies

Description	EXP. NO. 1	EXP. NO. 2	EXP. NO. 3	EXP. NO. 4	Experiments Average
Overall Accuracy	97.63	98.02	96.06	92.17	**95.97**
Classification Error	2.37	1.98	3.94	7.83	**4.03**
Detection Rate	97.24	97.70	95.41	90.86	**95.30**
Recall	97.63	98.02	96.06	92.17	**95.97**
Precision	96.67	97.22	94.44	88.89	**94.31**
F-measure	97.15	97.62	95.25	90.50	**95.13**

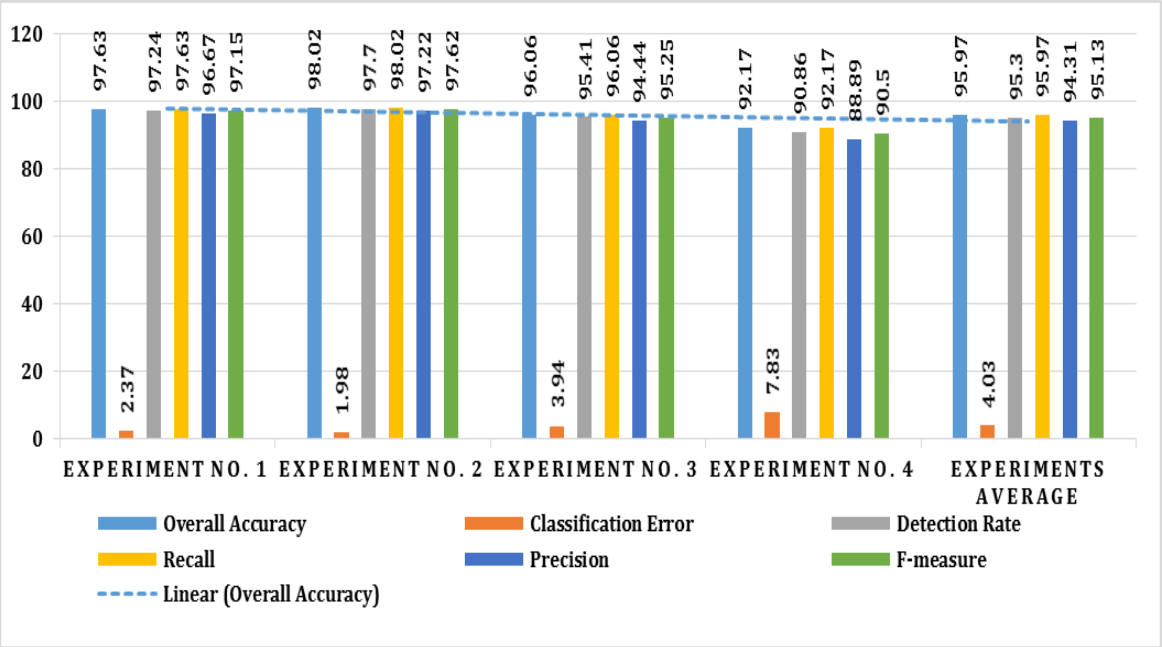

Figure 5.5: Results of the Performance Evaluation of Redundancy Anomalies

We note the result of the evaluation measuring is different in every experiment, these results are logical according to some factors as:

- The composition of dataset used in previous experiments, these dataset generate based different collection from subnetworks.
- The ordering of the rules is affect significantly in detection between the anomalous rules with normal rules.
- The degree of overlapping complexity between the anomalous and normal rules can affects significantly in evaluation results.

On other hand, the redundancy anomalous rules have a low degree of overlapping complexity between the rules, comparing with another type of the anomalous rules; the overlapping complexity between the rules depends as the following:

If every field of Rx is a superset or equal to the corresponding field in Ry and both rules have the same action.

Accordingly, the redundancy anomalous rules have the average of overall accuracy is 95.97% and this is a high average comparing with another type of the anomalous rules.

Summary

In this chapter, we presented four different experiment according to four type of anomalous rules as (shadowing anomalous, generalization anomalous, correlation anomalous and redundancy anomalous) have been constructed to detection the anomalous rules.

In experiments section, we applied in every experiment four different dataset sizes based on 70% normal rules and 30% anomalous rules and these dataset have a different composition according to collection from different subnetworks.

We performed these experiments using different overlapping relationship between the anomalous and normal rules based on each anomalous types, these dataset covered possible situations of anomalous rules.

On other hand, we noted the results for each experiment in order to evaluate and discuss these results.

In the evaluation section, we evaluated our model using confusion matrix to measuring the performance of detection the anomalous rules, Table 5.15 and Figure 5.6 illustrates the average summary of performance evaluation.

Table 5.15: Average Summary of Performance Evaluation

Anomalous Type	Overall Accuracy	Classification Error	Detection Rate	Recall	Precision	F-measure
Shadowing	91.82	8.18	90.42	91.82	88.33	90.03
Generalization	93.73	6.27	92.68	93.73	91.11	92.4
Correlation	89.31	10.69	87.47	89.31	84.72	86.81
Redundancy	95.97	4.03	95.3	95.97	94.31	95.13
Average Summary	**92.71**	**7.29**	**91.47**	**92.71**	**89.62**	**91.09**

Figure 5.6: Average Summary of Performance Evaluation

According to summarization in table 5.15 of evaluation average, we can extract some observations from it, the evaluation average is different results in every experiment, but it is logical results, because each experiment can affects by several factors as the follows:

- The composition of dataset used in every experiment is different, these dataset generate based different collection from subnetworks.

- The ordering of the rules is affect significantly in detection between the anomalous rules with normal rules.
- Every anomalous type have different degree of overlapping complexity between the anomalous and normal rules in detection can affects significantly in evaluation results.

Accordingly, the results of performance evaluation in detection the anomalous rules are logical and satisfaction, according to the overall accuracy average is 92.71%.

CHAPTER 6: Conclusion and Future Work

6.1 Research Conclusion

A firewall policy rules play core function in network security, however, the managing of firewall policy rules is very complicated task and error-prone. Therefore, the poor management of the firewall policy rules is resulting five type of anomalous rules, and this is one of the main reasons into vulnerability the network security.

To achieve our goal of our research based to the strengthening the accuracy factor in detection the anomalous rules. We applied different mechanism in matching process, through divided the IP address to four segments in array and matching every element in segment position with other element segment position in the same position and use the subnet mask to select the size of IP range.

In our research, we built sixteen different dataset divided to four groups based on four different experiments according to anomalous types, and was the average of previous experiments according to the overall accuracy is 92.71%. We believe that the result was acceptable based on the data used in the experiments.

6.2 Future Work

In the near future; the detection of anomalous firewall policy rules is still hot topics and research in networks security, however, the management of firewall rules has been proven to be complex task and error-prone**.** Therefore, the mismanagement of policy rules is very dangerous and main reasons in vulnerability of network security.

On the other hand, we will try to use down-top mechanism based on the matching between the anomalous rules and normal rules to increase detection the anomalous rules.

References

[1] Chapman, Brent and Elizabeth Zwicky, eds. 1995. Building Internet Firewalls (Second Edition). Cambridge: Orielly & Associates Inc.

[2] Han J., and Kamber M., "Data Mining: concepts and techniques", (2nd Edition), the Morgan Kaufmann Series in Data Management Systems, 2006.

[3] Olson D., and Delen D., "Advanced data mining techniques", Springer-Verlag Berlin Heidelberg, 2008.

[4] Ye N., "The Handbook Of Data Mining", Lawrence Erlbaum Associates, Inc., 2003.

[5] Al-Shaer E. and Hamed H. ,"Modeling and Management of Firewall Policies" Network and Service Management, IEEE Transactions, Vol.1, Issue: 1, pp.2-10, 2008.

[6] Al-Shaer E., Hamed H., Boutaba R and Hasan M., "Conflict classification and analysis of distributed firewall policies," IEEE Journal on Selected Areas in Communications, vol. 23, Issue: 10, pp. 2069-2084, 2005.

[7] Abedin M, Nessa S and Khan L," Detection and Resolution of Anomalous in Firewall Policy Rules", IFIP International Federation for Information Processing, Vol. 4127, pp. 15-29, 2006.

[8] Al-Shaer, E., Hamed, H., "Design and Implementation of Firewall Policy Advisor Tools", Technical Report CTI-techrep0801, School of Computer Science Telecommunications and Information Systems, DePaul University, 2002.

[9] Al-Shaer, E. and Hamed H., "Discovery of policy anomalous in distributed firewalls ", INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies, vol.4, pp.2605-2616, 2004.

[10] Anbarasan S., Balasubramani G, Madhan C, Naveenkumar P, Nithya5 S, "Detecting and Resolving Firewall Policy Anomalies Using Rule-Based Segmentation ",IJCSMC, Vol. 2, Issue. 4, pp. 134–137, 2013.

[11] Chaure R. and Shishir K. "Firewall anomalous detection and removal techniques – a survey ", International Journal on Emerging Technologies, vol.1, pp.71-74, 2010.

[12] Golnabi K. , Min, R.K.; Khan, L.; Al-Shaer, E, "Analysis of Firewall Policy Rules Using Data Mining Techniques", Network Operations and Management Symposium, 2006. NOMS 2006. 10th IEEE/IFIP, pp.305–315, 2006.

[13] Hamed, H., Al-Shaer, E., "Taxonomy of conflicts in network security policies ", Communications Magazine, IEEE, Vol.44, Issue 3, pp.134-141, 2006.

[14] Hongxin Hu, Gail-Joon Ahn and Ketan Kulkarni, "Detecting and Resolving Firewall Policy Anomalous " IEEE Transactions on Dependable and Secure Computing, vol. 9, issue 3, pp. 318-331, 2012.

[15] Jeffrey A. and Samak T., "Model Checking Firewall Policy Configurations," IEEE International Symposium on Policies for Distributed Systems and Networks, pp. 60-67, 2009.

[16] Golnabi K., Richard K. Min, Khan L. and, Al-Shaer E. ,"Analysis of Firewall Policy Rules Using Data Mining Techniques", Network Operations and Management Symposium 10th IEEE/IFIP, pp. 305–315, 2006.

[17] Khorchani B., Halle, S. and Villemaire, R. "Firewall anomaly detection with a model checker for visibility logic " Network Operations and Management Symposium (NOMS), 2012 IEEE, pp.466-469,2012.

[18] Luan. Y, Chen H., J. Mai, C. Chuah, Z. Su, Mohapatra P. and Davis C., "Fireman: A Toolkit for Firewall Modeling and Analysis," Proc. IEEE Symp. Security and Privacy, 2006.

[19] Pedditi S., Zhang Du, and Chung-E Wang, "FIEP: An Initial Design of A Firewall Information Exchange Protocol" IEEE 14th International Conference on Information Reuse and Integration (IRI), 2013.

[20] Khummanee S., Khumseela A. and Puangpronpitag S., "Towards a New Design of Firewall: Anomaly Elimination and Fast Verifying of Firewall Rules" 10th International Joint Conference on Computer Science and Software Engineering (JCSSE), pp. 93-98, 2013.

[21] Selvakanmani S. "A Novel Management Framework for Policy Anomaly in Firewall" International Journal for Scientific Research & Development| Vol. 1, Issue 9, pp.1710-1715, 2013.

[22] Yaz. B. , Mayer A. , Nissim K., and Firmato A., "A Novel Firewall Management Toolkit", In IEEE Symposium on Security and Privacy, pp.17-31, 1999.

[23] Scarfone K and Hoffman P, Guidelines on Firewalls and Firewall Policy, Recommendations of the National Institute of Standards and Technology, 2009.

[24] "Basic Flow Chart of Income and outgoing over Packet Filtering Firewall ", http://sanketrjain.com/what-are-firewalls-and-different-types-of-firewall [Accessed on 1/12/2014].

[25] "Description of Dataset attributes "http://docs.openstack.org/hot-reference/content/OS__Neutron__FirewallRule.html [Accessed on 1/12/2014].

[26] "Firewall IPTables Fundamentals ", http://www.thegeekstuff.com/2011/01/iptables-fundamentals/ [Accessed on 1/12/2014].

[27] "Code-Igniter framework ", http://www.codeigniter.com/userguide2/ [Accessed on 1/1/2015].

[28] "Linear search", http://en.wikipedia.org/wiki/Linear_search, [Accessed on /1/2015].

[29] "PHP 5 Tutorial ", http://www.w3schools.com/php/, [Accessed on 1/1/2015].