# Accessing Private Network via Firewall Based On Preset Threshold Value

[1]Shobharani Patil, [2]Bhavana. S, [3]Dr Jayashree. Agarkhed
*[1]MTech 4th sem Dept Of CSE VTU Regional PG Centre, Gulbarga*
*[2]Assistant Professor, Dept of CSE VTU Regional Center, Gulbarga.*
*[3]Dept of Computer Science and Engineering PDA College of Engineering Gulbarga, India*

***Abstract:*** *Many of the private services are made readily available to the users publically from emerging technologies such as web services, cloud computing, service-oriented architecture in an more effective and efficient manner. However, the users still suffer from security leakages by unauthorized one. Firewalls are the vital elements in network security. The worthiness of the firewall mainly depends on the quality of policies deployed in the firewall. Deploying and configuring firewall policies are more tedious and error-prone. In this Paper, we present an effective policy anomaly management framework for firewalls, anomalies are identified using rule-based segmentation technique and are derived effective anomaly resolution .In certain, we also make use of grid-based visualization technique, providing an intuitive cognitive sense about policy anomaly. User's requests are allowed to access the private network based on preset threshold value. In particular, we also demonstrate how effectively our approach can identify and overcome anomalies in firewall policies through various experiments.*

***General Terms:*** *Firewall, Policy anomaly management framework, Segmentation, Threshold value.*

## I. Introduction

As one of the essential elements in network security, firewalls are widely deployed in discovering suspicious traffic and unauthorized access to Internet-based business. Due to increasing threats in networks, firewalls are frontier defense for secure network against threats and unauthorized traffic by filtering unwanted network traffic coming from or going to the secured network. To implement a security policy in a firewall, connected servers defines a set of filtering rules that are derived from the organizational network security requirements.

Author Al-Shaer etal [1] presented that their firewall policies consists of conflicts even though several administrators maintained those policies. Wool [2] inspected firewall policies collected from different organization and indicated that all discovered firewall policies have security flaws. The process of implementing policies into the firewall is tedious and error-prone. Firewall Policy analysis tools, namely Firewall Policy Advisor [1] and FIREMAN[3], with the intension of detecting firewall conflicting policy anomalies have been introduced. Since policy anomalies are complex in nature, system administrators are facing problems in resolving those detected policy anomalies.

In this paper, we presented an anomaly management framework for firewalls which is based on rule-based segmentation technique. Based on this technique, a network packet space characterized by firewall policy can be partitioned into a set of disjoint packet space segments. Each segment is defined with unique set of firewall rules. This technique has been overcome from research problems such as network traffic measurement [4], firewall testing [5] and optimization [6].In particular we also deal with a grid-based visualization technique to diagnose policy anomaly information in an intuitive way. Filtering rules are identified using correlation, redundancy and shadowing method.

This paper is organized as follows, Section 2 related work, Section 3 proposed system, Section 4 system architecture, Section 5 results, Section 6 concludes this paper.

## II. Related Work

There exist a number of methods and tools in detecting and resolving firewall policies to assist system administrators. Al-Shaer and Hamed[1] proposed a tool called Firewall Policy Advisor to detect pairwise anomalies in firewall rules. Yuan et al [3] presented FIREMAN, a toolkit can only show that there is a misconfiguration between one rule and its preceding rules, but cannot accurately indicate all rules involved in an anomaly. FAME, overcomes all these limitations by conducting a complete anomaly detection and providing more accurate anomaly diagnosis information.

In [7] authors Lumeta and Fang [8] allow user queries to analyse and manage the firewall policies. They introduced lightweight firewall testing tools but could not provide a comprehensive examination of policy misconfiguration.

In [9] author Hari et al provided an algorithm for detecting and resolving conflicts in a general packet filter. But the algorithm can only detect a specific correlation conflict, and resolve the conflict by adding a resolving filter, which is not suitable for resolving conflicts identified in firewall policies.

In[10] author Fu et al examined conflict detection and resolution issues in IPSec policies, which is not directly applicable in firewall policy analysis. Also, there exist other related work to deal with a set of conflict resolution strategies for access control including Fundulaki and Marx [11], Jajodia et al. [12] and Li et al. [13]. These conflicts resolution mechanisms can be accommodated in our fine grained conflict resolution framework.

In existing system the policy anomalies are not only detected but also resolved based on rule reordering. The client request is allowed and denied to access the server based on preset threshold value. Here in this, the preset threshold value is 5. If the threshold value is <= 5 it is known as Lower Threshold [LT] and the action is allowed wherein if the threshold value is >5 it is known as Upper Threshold [UT] and the action is denied.

## III. Proposed System

Here in a Proposed System, the policy anomalies are detected and resolved in following explained steps:

- Automatic rule generation
- Correlation of Packet Space Segment
- Action Constraint Generation
- Rule Reordering
- Data Package

### 3.1 Automatic Rule Generation

Fig 1.depicts that when the client wants to send data packets to the network, some set of firewall rules should be satisfied to allow the packets. For this, network admin from different location allocate certain firewall rules to the server. Here we are generating the firewall rules automatically. We have process this, by taking certain specifications and constraints. The specification are taken and mapped randomly to generate the firewall rules. The rules are generated in the rule engine, the action happens when a client sends data packet to rule engine.

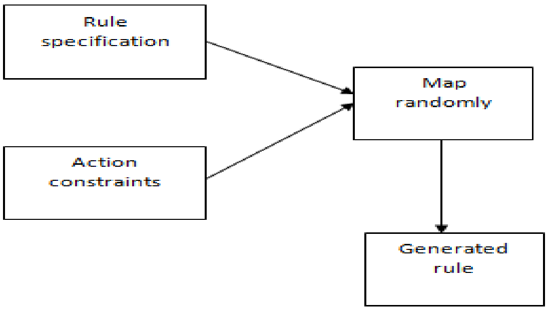


**Fig 1: Automatic rule Generation**

### 3.2 Correlation of Packet Space Segment

The major benefit of generating correlation groups for the anomaly analysis is that anomalies can be examined within each group independently, because all correlation groups are independent of each other which is in Fig 2. Especially, the searching space for reordering conflicting rules in conflict resolution can be significantly lessened and the efficiency of resolving conflicts can be greatly improved. In order to precisely identify policy anomalies and enable a more effective anomaly resolution, we introduce a rule-based segmentation technique. Therefore, one can utilize set operations to separate the overlapped spaces into disjoint spaces.

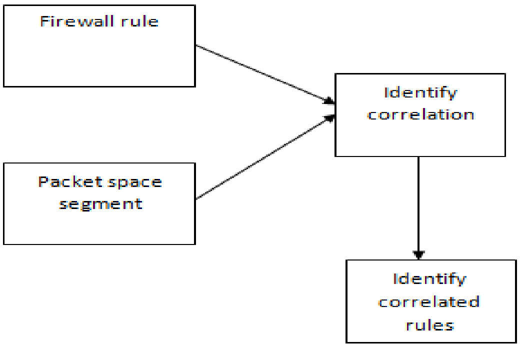


**Fig 2: Correlation of packet space segment**

### 3.3 Action Constraint Generation

In a firewall policy are discovered and conflict correlation groups are identified, the risk assessment for conflicts is performed. The risk levels of conflicts are in turn utilized for both automated and manual strategy selections. Fig 3 says a basic idea of automated strategy selection is that a risk level of a conflicting segment is used to directly determine the expected action taken for the network packets in the conflicting segment. If the risk level is very high, the expected action should deny packets considering the protection of network perimeters. On the contrary, if the risk level is quite low, the expected action should allow packets to pass through the firewall so that the availability and usage of network services cannot be affected. Thus, conflict resolution strategies (RS) can be generated automatically for partial conflict segments by comparing the risk levels with two thresholds, UT and LT, which can be set by system administrators in advance based on the different situations of protected networks.

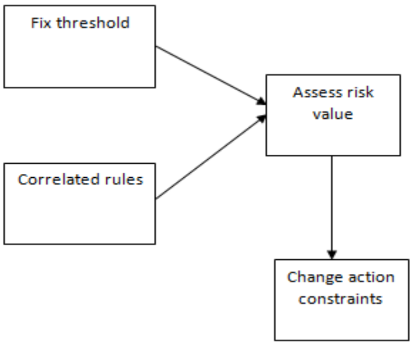


**Fig 3: Action Constraint Generation**

### 3.4 Rule Reordering

The solution for conflict resolution is that all action constraints for conflicting segments can be satisfied by reordering conflicting rules in Fig 4. In conflicting rules in order that satisfies all action constraints, this order must be the optimal solution for the conflict resolution. For all conflicting rules in a correlation group, our greedy conflicting resolution algorithm first calculates a resolving score for each conflicting rule individually. Then, the rule with the greatest resolving score is selected to solve the conflicts: a position range with the best conflict resolution is identified for the selected rule; and moving the selected rule to the new position achieves a locally optimal conflicting resolution.

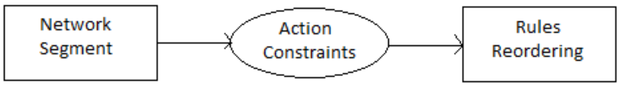


**Fig 4: Rule Reordering**

### 3.5 Data Package

When conflicts in a policy are resolved, the risk value of the resolved policy should be reduced and the availability of protected network should be improved comparing with the situation prior to conflict resolution based on the threshold value data will be received in to the server. Fig 5 explains about the evaluation of the risk reduction and availability improvement of conflict resolution approach, the results of conflict-resolved policies compared with the original policies as well as the best case and worst case with respect to the conflict resolution. The best case of a conflict resolution is achieved when all action constraints assigned to the conflicting segments can be satisfied. The worst case considering the security risk is that all packets covered by conflicting segments are allowed to pass through a firewall. And the worst case considering the availability is that all packets covered by conflicting segments assigned with "allow" action constraints are denied.

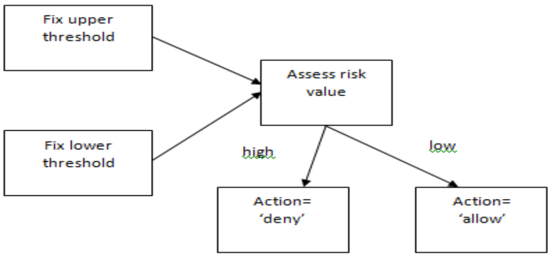


**Fig 5: Data Package**

## IV. System Architecture

Fig 6 shows that when a client sends a data packet to network, firewall checks the packet characteristics and decides to allow/deny the packet flow into the network. The firewall rule anomalies are identified using packet space segmentation technique, and then the risk of anomalies is assessed, based upon the risk, the firewall rules are re-ordering. Risk assessment is measured using a upper bound and lower bound threshold values.

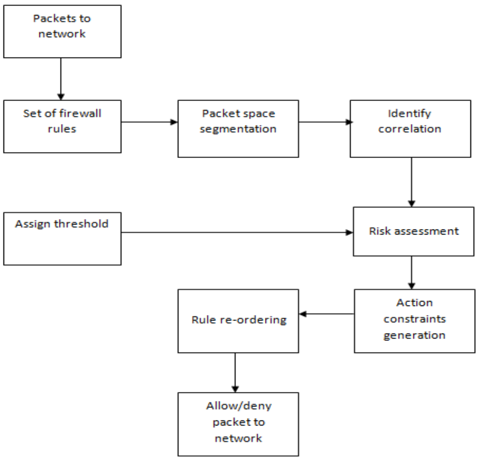
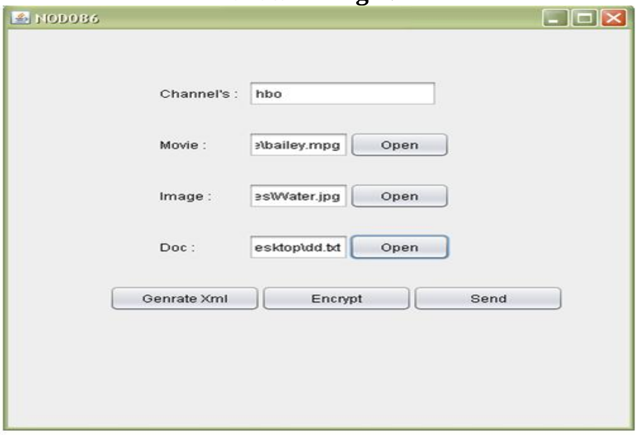


**Fig 6: Architecture of the packet access**

## V. Results

**Step 1 Client login into the channel and select file type like text, image, video and send to server which is shown in fig 7.**



Channel's : hbo

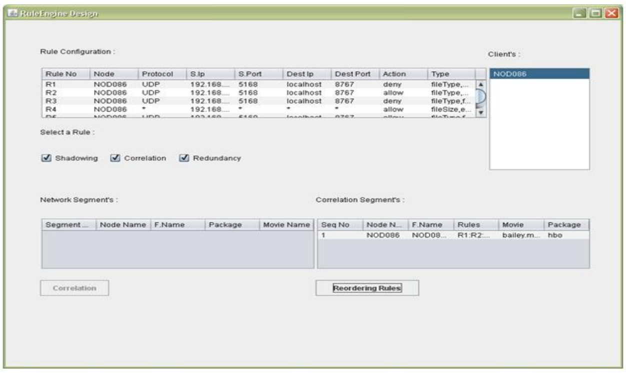
Movie : e\bailey.mpg

Image : es\Water.jpg

Doc : esktop\dd.txt

**Fig 7: Client Details**

**Step 2 In Firewall rule engine, certain random firewall rules are generated which is shown in fig 7. The correlation, redundancy and shadowing rules are identified.**



Rule Configuration :

| Rule No | Node | Protocol | S.Ip | S.Port | Dest Ip | Dest Port | Action | Type |
|---|---|---|---|---|---|---|---|---|
| R1 | NOD086 | UDP | 192.168.... | 5168 | localhost | 8767 | deny | fileType,... |
| R2 | NOD086 | UDP | 192.168.... | 5168 | localhost | 8767 | allow | fileType,... |
| R3 | NOD086 | UDP | 192.168.... | 5168 | localhost | 8767 | deny | fileType,f... |
| R4 | NOD086 | * | 192.168.... | * | * | * | allow | fileSize,e... |

Client's : NOD086

Select a Rule :

☑ Shadowing ☑ Correlation ☑ Redundancy

Network Segment's :

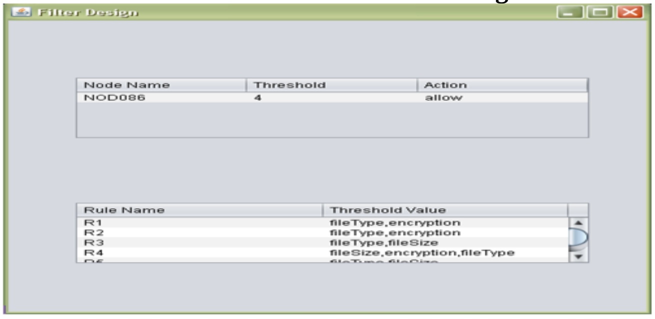
| Segment ... | Node Name | F.Name | Package | Movie Name |
|---|---|---|---|---|

Correlation Segment's :

| Seq No | Node N... | F.Name | Rules | Movie | Package |
|---|---|---|---|---|---|
| 1 | NOD086 | NOD08... | R1:R2:... | bailey.m... | hbo |

Fig 8: Rule Generation

**Step 3 When the assessed risk level is identified as LT then the packets are allowed through the network to access which is shown in below fig 8.**

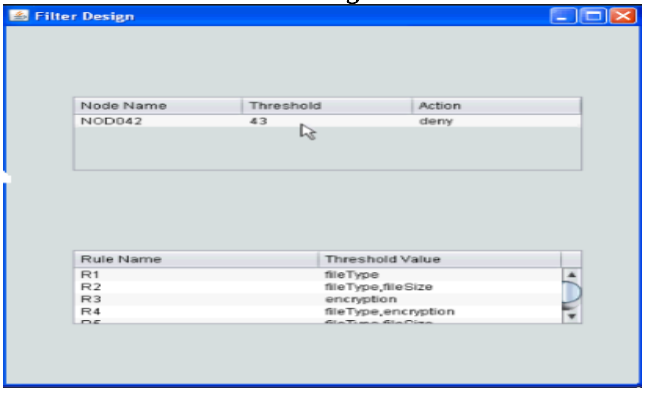


| Node Name | Threshold | Action |
|---|---|---|
| NOD086 | 4 | allow |

| Rule Name | Threshold Value |
|---|---|
| R1 | fileType,encryption |
| R2 | fileType,encryption |
| R3 | fileType,fileSize |
| R4 | fileSize,encryption,fileType |

Fig 8: Action Allow

**Step 4 When the assessed risk level is identified as UT then the packets are denied to transmit shown in below fig 9.**



| Node Name | Threshold | Action |
|---|---|---|
| NOD042 | 43 | deny |

| Rule Name | Threshold Value |
|---|---|
| R1 | fileType |
| R2 | fileType,fileSize |
| R3 | encryption |
| R4 | fileType,encryption |

Fig 9: Action Deny

## VI. Conclusion And Future Work

This paper concludes that, a novel anomaly management framework that facilitates systematic detection and resolution of firewall policy anomalies is proposed. A rule-based segmentation mechanism and a grid-based representation technique were introduced to achieve the goal of effective and efficient anomaly analysis. Anomaly management environment called FAME is proposed and demonstrated that proposed anomaly analysis methodology is practical and helpful for system administrators to enable an assurable network management.

Our future work extends our anomaly analysis approach to handle distributed firewalls. Moreover, we would explore how our anomaly management framework and visualization approach can be applied to other types of access control policies.

## References

[1]. E. Al-Shaer and H. Hamed, “Discovery of Policy Anomalies in Distributed Firewalls,” IEEE INFOCOM ’04, vol. 4, pp. 2605-2616, 2004.

[2]. A. Wool, “Trends in Firewall Configuration Errors: Measuring the Holes in Swiss Cheese,” IEEE Internet Computing, vol. 14, no. 4,
pp. 58-65, July/Aug. 2010

[3]. L. Yuan, H. Chen, J. Mai, C. Chuah, Z. Su, P. Mohapatra, and C. Davis, “Fireman: A Toolkit for Firewall Modeling and Analysis,” Proc. IEEE Symp. Security and Privacy, p. 15, 2006.

[4]. L. Yuan, C. Chuah, and P. Mohapatra, “ProgME: Towards Programmable Network Measurement,” ACM SIGCOMM Computer Comm. Rev., vol. 37, no. 4, p. 108, 2007.

[5]. A. El-Atawy, K. Ibrahim, H. Hamed, and E. Al-Shaer, “Policy Segmentation for Intelligent Firewall Testing,” Proc. First Workshop Secure Network Protocols (NPSec ’05), 2005.

[6]. G. Misherghi, L. Yuan, Z. Su, C.-N. Chuah, and H. Chen, “A General Framework for Benchmarking Firewall Optimization Techniques,” IEEE Trans. Network and Service Management, vol. 5,no. 4, pp. 227-238, Dec. 2008

[7]. A. Wool, “Architecting the Lumeta Firewall Analyzer,” Proc. 10th Conf. USENIX Security Symp., vol. 10, p. 7, 2001.

[8]. A. Mayer, A. Wool, and E. Ziskind, “Fang: A Firewall Analysis Engine,” Proc. IEEE Symp. Security and Privacy, pp. 17-189,2000

[9]. A. Hari, S. Suri, and G. Parulkar, “Detecting and Resolving Packet Filter Conflicts,” Proc. IEEE INFOCOM, pp. 1203-1212, 2000.

[10]. ] Z. Fu, S. Wu, H. Huang, K. Loh, F. Gong, I. Baldine, and C. Xu, “IPSec/VPN Security Policy: Correctness, Conflict Detection and Resolution,” Proc. Int’l Workshop Policies for Distributed Systemsand Networks (POLICY ’01), pp. 39-56, 2001.

[11]. I. Fundulaki and M. Marx, “Specifying Access Control Policies for XM L Documents with Xpath,” Proc. Ninth ACM Symp. Access Control Models and Technologies, pp. 61-69, 2004.

[12]. S. Jajodia, P. Samarati, and V.S. Subrahmanian, “A Logical Language for Expressing Authorizations,” Proc. IEEE Symp. Security and Privacy, pp. 31-42, May 1997.

[13]. N. Li, Q. Wang, W. Qardaji, E. Bertino, P. Rao, J. Lobo, and D. Lin, “Access Control Policy Combining: Theory Meets Practice,” Proc. 14th ACM Symp. Access Control Models and Technologies, pp. 135- 144, 2009.