Journal of Information Security, **2011, 2, 122-130**
doi:10.4236/jis.2011.23012 Published Online July 2011 (http://www.SciRP.org/journal/jis)

Proactive Security Mechanism and Design for Firewall

Saleem-Ullah Lar[1,3], Xiaofeng Liao[2], Aqeel ur Rehman[1], Qinglu Ma[1]
[1]*Department of Computer Science, Chongqing University, Chongqing, China*
[2]*Faculty of Computer Science, Senior Member IEEE, Chongqing University, Chongqing, China*
[3]*Department of Computer Science and IT, The Islamia University Bahawalpur, Pakistan*
E-mail: *salimbzu@gmail.com*

Received June 2, 2011; *revised July* 5, 2011; *accepted July* 15, 2011

Abstract

In this paper we have present the architecture and module for internet firewall. The central component is fuzzy controller while properties of packets are fuzzified as inputs. On the basis of proposed fuzzy security algorithm, we have figured out security level of each packet and adjust according to packets dynamic states. Internet firewall can respond to these dynamics and take respective actions accordingly. Therefore, proactive firewall solves the conflict between speed and security by providing high performance and high security. Simulation shows that if the response value is in between 0.7 and 1 it belongs to high security.

Keywords: Firewall Security, Security Evaluation, Network Security

1. Introduction

The expansion of the Internet and e-Commerce has made organizations more vulnerable to electronic threats than ever before. With the increasing quantity and sophistication of attacks on IT assets, companies have been suffering from breach of data, loss of customer confidence and job productivity degradation, all of which eventually lead to the loss of revenue. According to the 2004 CSI/FBI Computer Crime and Security survey [1], organizations that acknowledged financial loss due to the attacks (269 of them) reported $141 million lost, and this number has only grown since. Moreover, as unskilled, unmanned attacks such as worms and viruses multiply the probability of attack approaches for every organization. The question therefore shifts from whether an attack will occur, to when an attack will occur. Thus, a sound IT security plan is more important than ever, and the protection provided by current and emerging Intrusion Prevention Systems (IPS) is becoming a critical component [2-5].

IPS utilizes IDS algorithms to monitor and drop or allow traffic based on expert analysis. These devices normally work at different areas in the network and proactively monitor any suspicious activity that could otherwise bypass the firewall. IPS “firewalls” can intelligently prevent malicious traffic from entering/exiting the firewall and then alert administrators in real time about any suspicious activity that may be occurring on the network [6]. A complete network IPS solution also has the capability to enforce traditional static firewall rules and administrator-defined whitelists and blacklists.

Though IPS devices are the most resource intensive, they are still relatively high-performing due to the latest processors, software, and hardware advancements. IPS may be distributed and hardware based [7-10]. Today two categories of IPS exist: Network-based Intrusion Prevention and Host-based Intrusion Prevention. Network IPS monitors from a network segment level, and can detect and prevent both internal and external attacks. Network IPS devices separate networks in much the same fashion as firewalls. Host IPS software runs directly on workstations and servers detects and prevents threats aimed at the local host. In both cases, attack recognition is usually accomplished via two primary methods of IDS: known-attack detection, and anomalous behavior detection.

This paper focuses on fuzzy mechanism with the help of Gaussian mechanism as a member function and center of gravity procedure which is an implementation of a fuzzy inputs and outputs respectively in the model. The rest of the paper is organized as follows: Section 2 presents the challenges faced by traditional security architectures. Section 3 describes proposed firewall architecture. Section 4 explains about proposed proactive fuzzy security mechanism. Finally, Section 5 presents simulation results and concludes the paper.

2. The Challenges for Traditional Security Architecture

In fact, it is still the Firewall that plays the key role in traditional security architecture, since it controls most of the incoming and outgoing traffic of an enterprise. Essentially the firewall is almost a must-have in each enterprise. To review the challenges for the traditional architecture, undoubtedly it is necessary to address on the limitation of traditional firewalls. The inability of current firewalls may include:

1) Limited ports & performance.

2) Complicated UI configuration and policy management.

3) Scalability limitation to correspond to organization growth.

4) Unreliable network security, due to "Single Point of Defense.

5) Insufficient capability to effectively manage emerging internet applications hidden in HTTP traffic.

6) Passive security mechanism to respond network threats including network worms, Trojans and cyber-attacks.

Facing the emerging malicious codes, network worms and hybrid attacks today, traditional firewall is no longer effectively to harden your enterprise network. Traditional firewalls usually inspect the incoming traffic cautiously, and it can base on the network policies to permit, deny or drop the traffic depending on the traffic trusty or illegal. But for the outgoing traffic, unfortunately the HTTP traffic is always permitted in the enterprise network, and the firewalls are lack of the management capability to inspect the evolving internet applications which now can hide themselves in the HTTP traffic and sneak out. Thus, the enterprises gate seems secure but in fact, the security cracks have been created.

3. Proposed Firewall Architecture

A true firewall is the hardware and software that intercepts the data between the Internet and your computer. All data traffic must pass through it, and the firewall allows only authorized data to pass into the corporate network. Firewalls are typically implemented using one of four primary architectures.

- Packet Filters
- Circuit-level Gateways
- Application Proxies
- Network Address Translation

3.1. Definition

Our definition covers the state of firewall technology as a distributed security architecture placed on the data transmission path between communication endpoints. Our definition of firewall technology states that communication traffic needs to enter or leave a network security domain to be of interest to firewall technology. **Figure 1** illustrates the possible combinations for point-to-point communication. For any traffic between sender ***a_i*** and receiver ***b_i*** the definition includes traffic that traverses the protected domain $D_A(\{a_i, b_i\} \notin D_A,\ i=1)$ and traffic that traverses networks that are not part of $\boldsymbol{D_A}$ with $\boldsymbol{a_{i\varepsilon}D_A}$ and $\boldsymbol{b_i}$ $\boldsymbol{D_A}$ (outbound traffic; i = 2), $\boldsymbol{a_i}$ 2= $\boldsymbol{D_A}$ and $\boldsymbol{b_i}$ 2 $\boldsymbol{D_A}$ (inbound traffic i = 3), or both $\boldsymbol{a_i}$ 2 $\boldsymbol{D_A}$ and $\boldsymbol{b_i}$ 2 $\boldsymbol{D_B}$ (virtual private networking between $\boldsymbol{D_A}$ and $\boldsymbol{D_B}$; i = 4). Communication traffic between $\boldsymbol{a_i}$ and $\boldsymbol{b_i}$ that neither enters nor leaves a network policy domain is not subject to firewall technology.

$$\begin{bmatrix} \text{Sender} & a_i \\ \text{Receiver} & b_i \end{bmatrix} i\varepsilon\{1,2,3,4\}$$

$$a_i \varepsilon D_A \left(\{a_i, b_i\} \notin D_A,\ i=1\right)$$

Fuzzy agent is the basic element in this architecture specific attack or a particular phase of an attack. It consists of three components; fuzzy Context, exponential moving average module and fuzzy inference engine shown in **Figure 2**. Fuzzy context represents the problem domain *i.e.* normal profile of network in reference to particular intrusion. Exponential moving average module adapts the fuzzy context according to current network conditions and traffic patterns, while fuzzy inference engine actually classifies an event using fuzzy knowledge base and real-time inputs. Fuzzy context is a key component of the fuzzy agent, which consists of rules and membership functions. Context generation and evolution module constructs optimized rules and membership functions for current network. Fuzzy rules can be expressed in terms of simple if-then statements with higher interpretability score. Let the fuzzy sets for fuzzy input variables are low, medium and high. The membership functions of each linguistic fuzzy set in terms of boundary parameters are describe by Equations (1)-(2). The boundary parameters are functions of evolved parameters as defined in Equation (5) and moving average

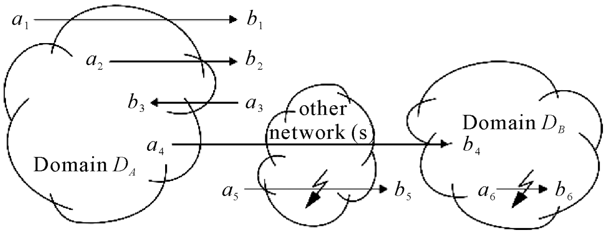


Figure 1. Communication traffic governed by firewall technology between senders and receivers.

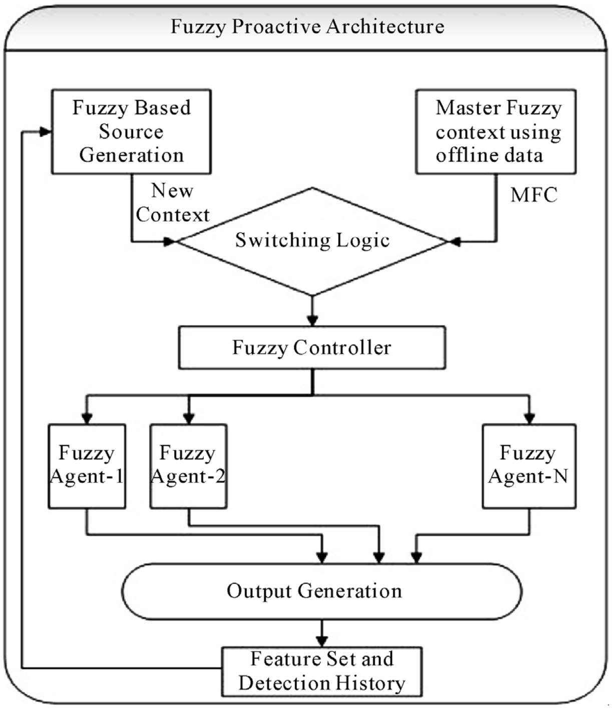

Fuzzy Proactive Architecture
Fuzzy Based Source Generation
Master Fuzzy context using offline data
New Context
MFC
Switching Logic
Fuzzy Controller
Fuzzy Agent-1
Fuzzy Agent-2
Fuzzy Agent-N
Output Generation
Feature Set and Detection History

Figure 2. Architecture of a fuzzy typical approach.

modules output. Member-ship functions contract or expand linearly according to network history depending upon exponential moving average modules output. This helps in adjusting the attack threshold value at that particular interval while evolved parameters set the normal and not-normal class boundaries.

Fuzzy inference engine that is third component of fuzzy agent, classifies the real-time input as normal or malicious using fuzzy knowledge base. It basically accomplishes three functions (fuzzification, fuzzy inference, defuzzification) based on Mumdani principle [11]. In fuzzification, a crisp input *i.e.* a record from feature set is mapped to fuzzy sets to determine the membership degree. The inference engine evaluates applicable rules and their degree of matching to generate consequent rules. The defuzzification function aggregates the consequent rules and using centroid method, generates one crisp output, which determines the class of input record [11].

3.2. Controller

Proposed mechanism is employed in the controller which is the core module this firewall. The controller has the functionality to integrate with the arrival packets (inputs) applied rules, and fuzzy logic to measure the security level of arriving packets. Using these values controller has to do following main tasks to process the connections accordingly.

1) Filtration
2) Dynamic Monitoring

3.3. Dynamic Packet Filtering

Dynamic packet filtering is a firewall and routing capability that provides network packet filtering based not only on packet information in the current packet, but also on previous packets that have been sent. For example without dynamic packet filtering, a connection response may be allowed to go from the internet to the secure part of the network. Dynamic packet filtering would consider whether a connection was started from inside the secure part of the network and only allow a connection response from the internet if the packet appeared to be a response to the request.

Dynamic packet filtering filters packets based on:

1) Administrator defined rules governing allowed ports and IP addresses at the network and transport layers of the OSI network model.

2) Connection state which considers prior packets that have gone through the firewall.

3) Packet contents including the application layer contents

Static packet filtering only filters packets based on administrator defined rules governing allowed ports and IP addresses at the network and transport layers of the OSI network model as mentioned in item 1 above. Therefore dynamic packet filtering also called state-full inspection which provides additional capabilities including inspection of packet contents up to the application layer and consideration of the state of any connections.

Dynamic packet filtering provides a better level of security than static packet filtering since it takes a closer look at the contents of the packet and also considers previous connection states.

3.4. Network Address Translation

NAT is a very important aspect of firewall security. It conserves the number of public addresses used within an organization, and it allows for stricter control of access to resources on both sides of the firewall. Most modern firewalls are state full—that is, they are able to set up the connection between the internal workstation and the Internet resource. They can keep track of the details of the connection, like ports, packet order, and the IP addresses involved. This is called keeping track of the state of the connection. In this way, they are able to keep track of the session composed of communication between the workstation and the firewall, and the firewall with the Internet. When the session ends, the firewall discards all of the information about the connection. It is suggested to design network using RFC-1918 [12] that never advertised outside from the intranet. The mapping is dynamic so it is difficult to guess either two connections with the same IP actually come from the same or different hosts.

3.5. Security Rules and Policies

Allowing or denying services or connections between networks defined by security policies and rules.

4. Proactive Fuzzy Security Mechanism

Saniee Abadeh [13] presents combined fuzzy logic and genetic algorithm to evolve fuzzy rules, optimize membership functions to detect new anomalies. While our proposed proactive firewall security mechanism which is employed in the fuzzy controller is different and explained as follows.

4.1. Proactive Control

Since the state of packets in the networks is constantly varying, its security level is also changeable. Previous secure user may initiate malicious attack or disobey the security rules. So the fields of "attack times" are used to record the times of disobeying security rules. Accordingly, the source or destination security values will be adjusted to respond to its varying security state. When the source and destination security vary from 1 to 0, the overall security level of the connection smoothly vary accordingly. Therefore, the output can reflect the changes of packets status. Different methods and security policies are used for 1148 different kinds of connections and policies of control over them are adjusted according to their varying states. So, the firewall is fuzzily adaptive and proactive.

4.2. Source Generation

Figure 3 describe Input generation based on source and destination security values employed in fuzzy controller. Range of input is [0, 1] and value is directly proportional to security level. We have defined Gaussian member function for the source security, which is represented as

$$\mu_S(s,c,\sigma) = \mathrm{e}^{\frac{(s-c)^2}{2\sigma^2}} \quad 0 \le S \le 1 \tag{1}$$

μ_{Sl}, μ_{Sm} and μ_h denoted as Low, Medium, and High security levels for the source member function respectively depending on parameters σ and $\boldsymbol{c}$.

$$\mu_D(D,c,\sigma) = \mathrm{e}^{\frac{(D-c)^2}{2\sigma^2}} \quad 0 \le D \le 1 \tag{2}$$

μ_{Dl}, μ_{Dm} and μ_{Dh} denoted as Low, Medium, and High security levels for the destination member function respectively.

4.3. Applied Rules and Regulations

For our system we have defined the rules as shown in the **Figure 3**, while fuzzy applied relations for the applied rules are as follows.

***Rule* 1** IF source = low and destination = low
THEN security = low

***Rule* 2** IF source = low and destination = medium THEN security = low

Rule n IF source = high or destination = high
THEN security = high

Mathematically we can define applied relations as,

For Rule 1: $\boldsymbol{\mu}_{R1} = \boldsymbol{\mu}_{S1} \cap \boldsymbol{\mu}_{D1} \cap \boldsymbol{\mu}_{Z1}$

For Rule *n*: $\boldsymbol{\mu}_{Rn} = \boldsymbol{\mu}_{Sn} \cap \boldsymbol{\mu}_{Dn} \cap \boldsymbol{\mu}_{Zn}$

So we can write that,

$$\boldsymbol{\mu}_R = \boldsymbol{\mu}_{R1} \cap \boldsymbol{\mu}_{R2} \cap \boldsymbol{\mu}_{Rn} \tag{3}$$

Therefore,

$$Z = (S * D) \cdot R \tag{4}$$

and

$$\boldsymbol{\mu}(z) = \cup\left[\boldsymbol{\mu}_S \cap \boldsymbol{\mu}_D \cap \boldsymbol{\mu}_R\right] \tag{5}$$

We defined above rules just to cope up with the issue of input space up to maximum possible effort. Since process mostly requires non-fuzzy values, so defuzzification process is necessary to implement this is described in next section. For low priority based trusted packets both application level and dynamic packet monitor are used providing high security, while filtration takes place for highly trusted packets. It is fuzzily adaptive and proactive in a sense that its characteristics and packet status are fuzzified and its output reflects the packet dynamic status (**Figure 4**).

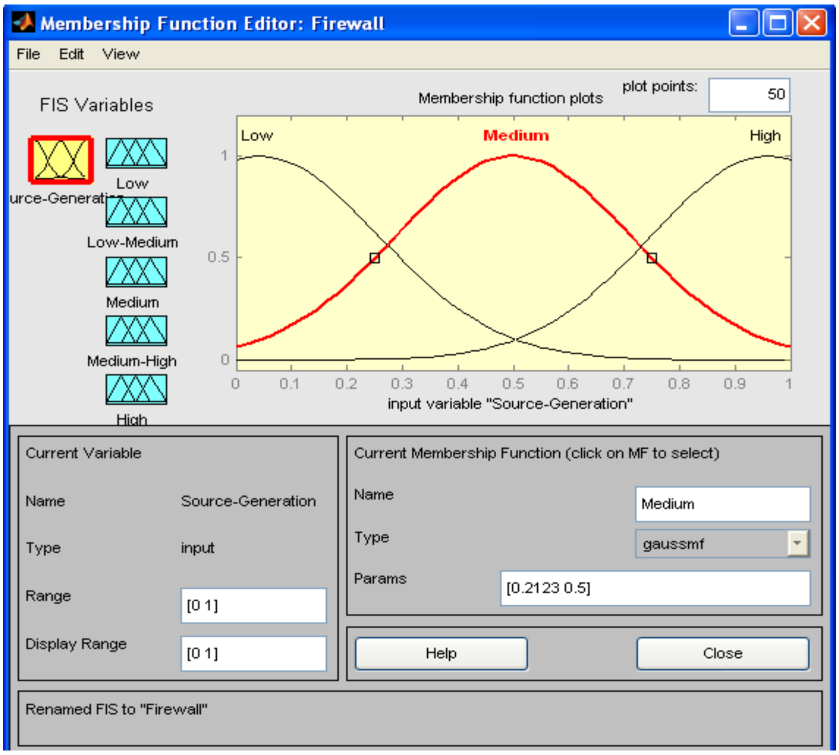
4.4. Destination Generation

We have defined member function for destination output which is obtained from Equation (5) as,

$$Z_0 = \frac{\int_z z\mu(z)\,\mathrm{d}z}{\int_z \mu(z)\,\mathrm{d}z}$$

The above equation used is based on center of gravity method.

Figure 5 shows the characteristics and security level designed for output generation based on the rules and relations described earlier.



plot points: 50

Current Variable

Name Source-Generation

Type input

Range [0 1]

Display Range [0 1]

Current Membership Function (click on MF to select)

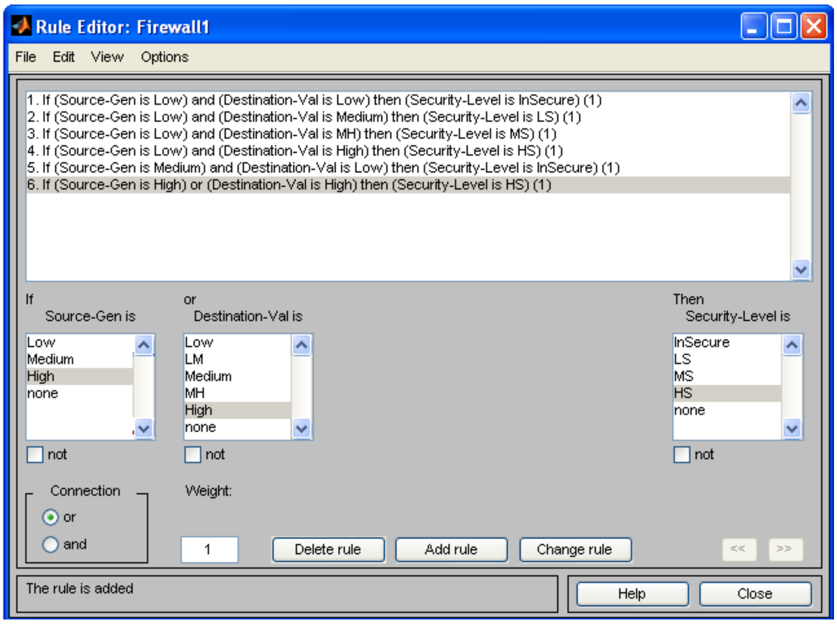
Name Medium

Type gaussmf

Params [0.2123 0.5]

Renamed FIS to "Firewall"

Figure 3. Input members function generation.



1. If (Source-Gen is Low) and (Destination-Val is Low) then (Security-Level is InSecure) (1)
2. If (Source-Gen is Low) and (Destination-Val is Medium) then (Security-Level is LS) (1)
3. If (Source-Gen is Low) and (Destination-Val is MH) then (Security-Level is MS) (1)
4. If (Source-Gen is Low) and (Destination-Val is High) then (Security-Level is HS) (1)
5. If (Source-Gen is Medium) and (Destination-Val is Low) then (Security-Level is InSecure) (1)
6. If (Source-Gen is High) or (Destination-Val is High) then (Security-Level is HS) (1)

Connection ● or ○ and

Weight: 1

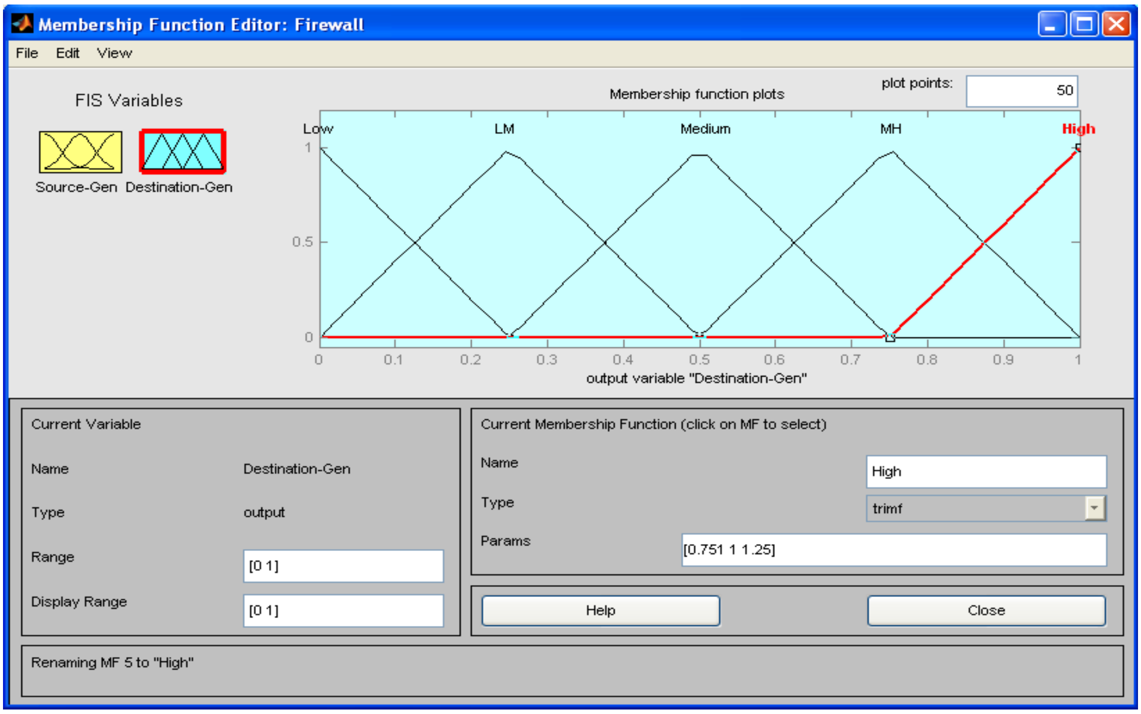
The rule is added

Figure 4. Defining fuzzy rules.

5. Simulation and Analysis

This section describes the experimental results and performance evaluation of the proposed system. The proposed system is implemented in MATLAB (7.0.1). Based on above defined procedure our simulation results described in the following figures. **Figure 6** describes the value generated by source and destination with its security level based on the defined rules. We can see that values on both sides are almost directly proportional which reflects the level of the security

The fuzzy rules given to the fuzzy system is done


Membership Function Editor: Firewall
File
Edit
View
FIS Variables
Source-Gen
Destination-Gen
Membership function plots
plot points:
Low
LM
Medium
MH
High
1
0.5
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
output variable "Destination-Gen"
Current Variable
Current Membership Function (click on MF to select)
Help
Close

plot points: 50

Name: Destination-Gen

Type: output

Range: [0 1]

Display Range: [0 1]

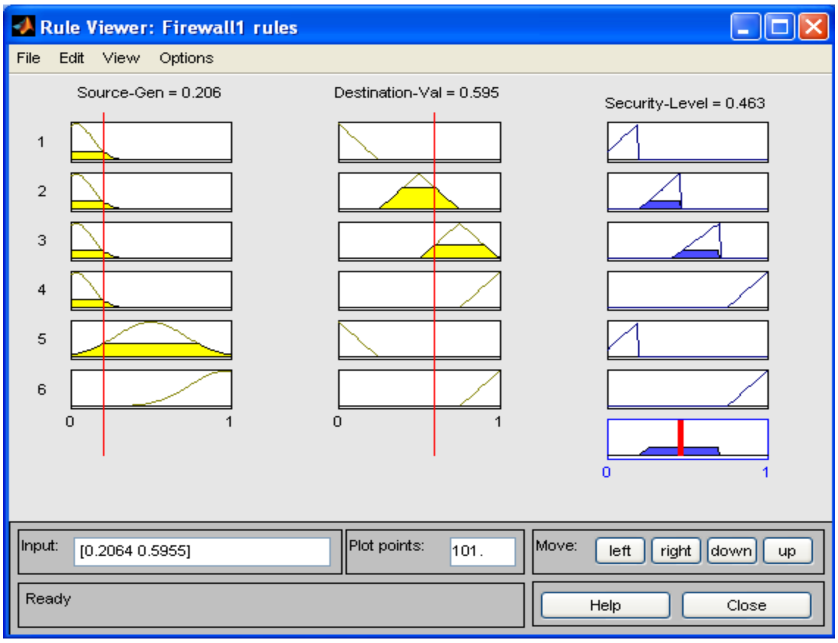
Name: High

Type: trimf

Params: [0.751 1 1.25]

Renaming MF 5 to "High"

Figure 5. Members function for destination output security.



Source-Gen = 0.206

Destination-Val = 0.595

Security-Level = 0.463

Input: [0.2064 0.5955]

Plot points: 101

Ready

Figure 6. Visualization of Source and destination with security level (rule observer).

manually by analyzing intrusion behavior. Some time it is very difficult to generate fuzzy rules manually due to the fact that the input data is huge and also having more attributes. But, a few of researches are available in the literature for automatically identifying of fuzzy rules in recent times. Motivated by this fact, we make use of mining methods to identify a better set of rules.

Table 1 and **Figure 7** shows the clear view about the security level for each connection.

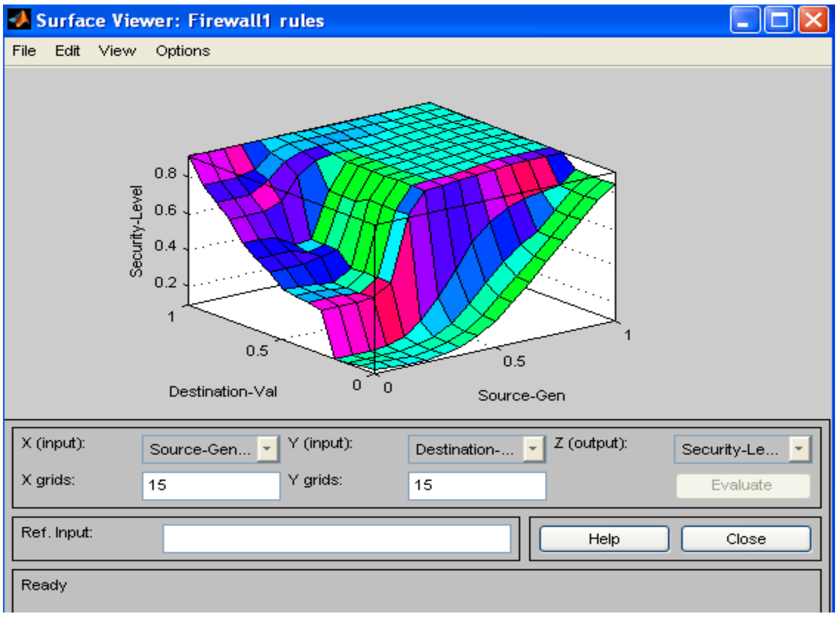
Various control method used to monitor and control the connection according to its security level. Therefore firewall is proactive, intelligent and remains secure and provide high performance.

A smoothly varying surface can provide the value of overall security level for each connection. It has been observe deeply through ramp function that input and output security varies from 0 to 1 and the overall security level also varies smoothly, and we can get the status of the packets from the output generation. The ramp function is an elementary unary real function, easily computable as the mean of its independent variable and its absolute value and it is derived by the look of the graph.

From **Figures 8** and **9** we can see that as source generated value increases or decreases it has clear effect on the security level and a particular action will be taken place based on the results.

6. Conclusions

In this work, fuzzy based system was designed to evaluate the threat level of identified threats, because it is impossible to provide assurance for the system and justify security measures incorporated unless the system is analyzed during the designing state of computer based systems. With this system designed, risk analysis has been made easier to perform.



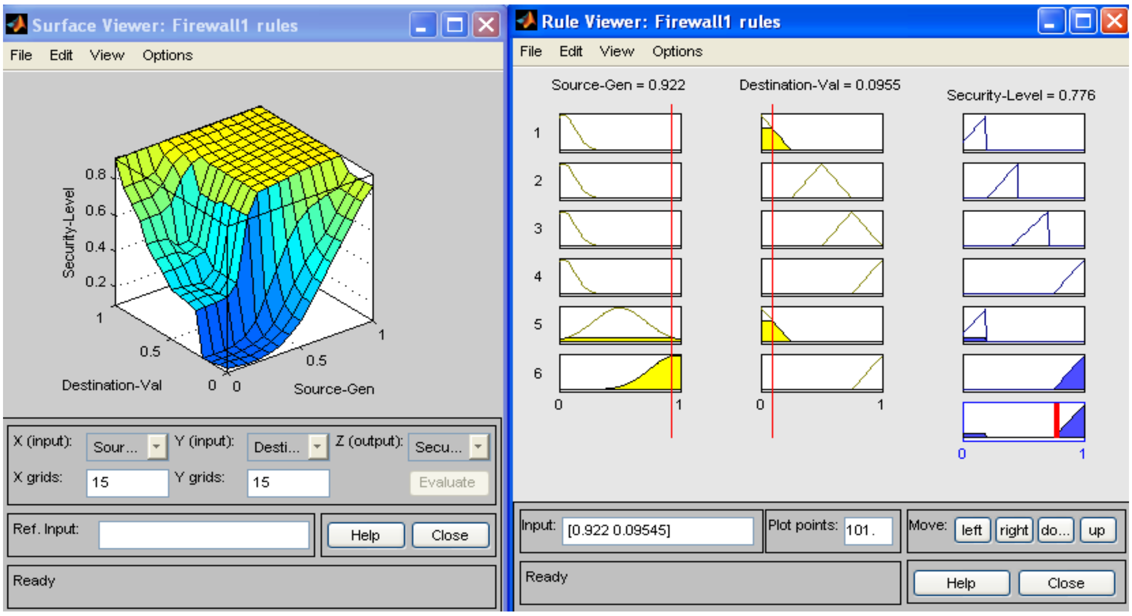
X (input): Source-Gen... Y (input): Destination-... Z (output): Security-Le...

X grids: 15 Y grids: 15

Ref. Input:

Ready

Figure 7. Surface level view (final result).



X (input): Sour... Y (input): Desti... Z (output): Secu...

X grids: 15 Y grids: 15

Ref. Input:

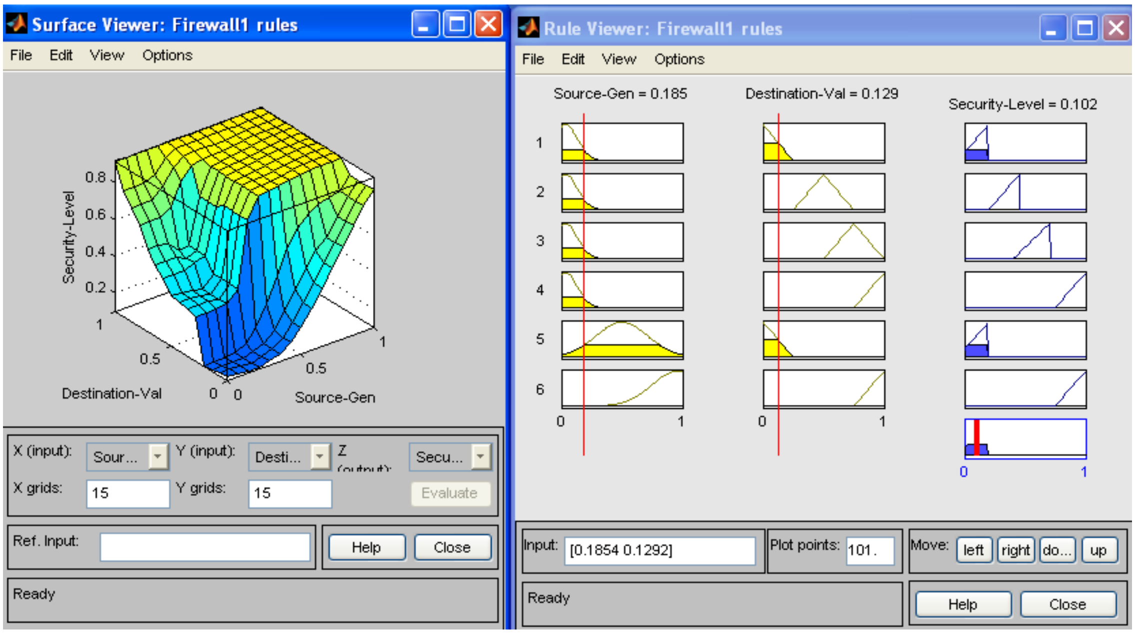
Ready

Source-Gen = 0.922 Destination-Val = 0.0955 Security-Level = 0.776

Input: [0.922 0.09545] Plot points: 101

Ready

Figure 8. Rule and surface viewer (high security).



X (input): Sour...

Y (input): Desti...

Z (output): Secu...

X grids: 15

Y grids: 15

Ref. Input:

Ready

Source-Gen = 0.185

Destination-Val = 0.129

Security-Level = 0.102

Input: [0.1854 0.1292]

Plot points: 101

Ready

Figure 9. Rule and surface viewer (low security).

Table 1. Security level for each connection.

Output Value -$\mu(z)$	Security Level	Action Taken for Connection
>0 and <0.2	Insecure	Denied
>0.2 and <0.4	Low Security	Dynamic Monitoring and Auditing
>0.4 and <0.7	Medium Security	Dynamic Monitoring and Filtering
>0.7 and <1	High Security	Only Filter

Overall security level and methods to control packets and connections can be adjusted as per network dynamic status. It resolves the issues between security and speed providing high security and high performance. It is fuzzily adaptive and intelligent and has flexibility with a high degree of performance.

7. Future Work

For further research, this system designed can be redesigned using object orientated programming language and other models like DREAD and SWOT model can be used.

8. References

[1] CSI/FBI, “Computer Crime and Security Survey,” 2004. http://i.cmpnet.com/gocsi/db_area/pdfs/fbi/FBI2004.pdf

[2] C. Baumrucker, J. Burton, S. Dentler, *et al*., “Cisco Security Professional’s Guide to Secure Intrusion Detection Systems,” Syngress Publishing, Burlington, 2003.

[3] C. Endorf, E. Schultz and J. Mellander, “Intrusion Detection & Prevention,” McGraw-Hill, Boston, 2004.

[4] “Technical Overview of The Bouncer,” http://www.cobrador.net/docs/whitepaper.pdf

[5] M. Barkett, “Intrusion Prevention Systems,” NFR Security, Inc., 2004. http://www.nfr.com/resource/downloads/SentivistIPS-WP.pdf

[6] K. Xinidis, K. G. Anagnostakis and E. P. Markatos, “Design and Implementation of a High Performance Network Intrusion Prevention System,” *Proceedings of the* 20*th International Information Security Conference* (*SEC* 2005), Makuhari-Messe, Chiba, 30 May-1 June 2005.

[7] T. Sproul and J. Lockwood, “Wide-Area Hardware-Accelerated Intrusion Prevention Systems (WHIPS),” *Proceedings of the International Working Conference on Active Networking* (*IWAN*), Lawrence, 27-29 October 2004.

[8] D. Sarang, K. Praveen, T. S. Sproull and J. W. Lockwood, “Deep Packet Inspection Using Parallel Bloom Filters,” *IEEE Micro*, Vol. 24, No. 1, 2004., pp. 52-61.

[9] D. V. Schuehler, J. Moscola and J. W. Lockwood, “Architecture for a Hardware-Based, TCP/IP Content-Processing System”, *IEEE Micro*, Vol. 24, No. 1, 2004, pp. 62-69.

[10] H. Song and J. W. Lockwood, “Efficient Packet Classification for Network Intrusion Detection Using FPGA,” *Proceedings of the International Symposium on Field-Programmable Gate Arrays* (*FPGA*’05), Monterey, 20-22 February 2005.

[11] J. Yen and R. Langari, “Fuzzy Logic: Intelligence, Control and Information,” Prentice Hall, Upper Saddle River, 1999.

[12] http://tools.ietf.org/html/rfc1918

[13] M. S. Abadeh, J. Habibi and C. Lucas, “Intrusion Detection Using a Fuzzy Genetics-Based Learning Algorithm,” *Journal of Network and Computer Applications*, Vol. 30, No. 2007, 2007, pp. 414-428.