IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified

Vol. 5, Issue 12, December 2016

A Study on Home Office Firewall

Krishna Varha Chaitanya Chintalapudi[1], P Ravi Kiran Varma[2]

M. Tech, Computer Networks and Information Security, MVGR College of Engineering, Vizianagaram, India [1]

CSE Department, MVGR College of Engineering, Vizianagaram, India[2]

Abstract: With the advancement in networking services, network security has become the key facet in our digital world. The term network security constitutes prevention of network and network resources by ceaseless monitoring and thereby denying the services like user access to network resources and modification of network configuration details from unauthorized users. Firewalls are one such network security system that monitors the inbound and outbound traffic of the network, this observation relies on two principal factors, Stateless and Stateful firewalls. The Stateless firewalls track the source and destination address of every incoming and outgoing data packets whereas the Stateful firewalls track each packet in the network by reading the entire header and thus it can read the type of data being transmitted. “PLUTO – A home office firewall” is a stateless firewall which can be used in Linux environment. The main motto behind developing the PLUTO as real time firewall is to demonstrate a real-time application for home office users through which they can secure their network.

Keywords: Pluto, home office firewall, firewalls, network perimeter security.

I. INTRODUCTION

Firewall and Intrusion Detection System (IDS) are the most important components of network perimeter security. [1]. Firewalls are one such network security systems that stand as a barrier through which the entire network traffic is to be passed. A firewall security policy describes which traffic is to be accepted into the network and which traffic is to be denied [2].

In general a firewall can filter the packets at IP packets similarly if needed firewalls can also be implemented at higher layers of the communication system. Firewalls are generally made to allow authorized traffic (packets that satisfy all the firewall security rules) into the network and deny rest of the traffic.
PLUTO allows the system administrators to define rules for the treatment of incoming and outgoing data packets of the network. This tool uses a tree-based approach replacing existing listed rule.

I.1 PLUTO – A home office firewall

Firewalls are barriers between us and rest of the world and PLUTO is one such firewall. we know that computers are connected to the internet by ports [3] (like port number 80 for HTTP traffic, port number 110 for receiving mails-pop3, port number 25 is typically used for sending mail-SMTP). The basic idea behind the implementation of firewall PLUTO is to close the unnecessary ports, if such ports are not closed, then there is no doubt in saying that this is an open invitation for hacking incidents.

I.2 Different types of threats for home office users

a. Hacking into the network through open parts: If in a network open port exists then it’s known as a security breach due to which someone can intrude into our network and may cause information loss.

b. Hacking as harmless network interaction in our network: If we don’t have any firewall in our network crackers may intrude into the network and may alter the networking devices configurations due to which our network devices may not perform the appropriate tasks (i.e. the intended task for which the device is made or purchased), as a result even the harmful activities in and on the network may be shown as harmless network communication.

c. Spoofed programs
A firewall generally detects spoofed programs which are generally made to spoof some or any useful resources and thereby make an attack (Ex. Spoofed antivirus tools which are originally made to extract information from the network).

d. Installing unknown spyware
Due to the absence of firewall the intruders may install any spyware and there by the intruder can extract information like user name and their respective password of the user in the network.

e. Scanning the ports and security breaches
Usually firewalls can scan the entire network for the available unnecessary open ports and security breaches inside the network to protect them from the outside world thus the absence of firewall usually help the intruders to intrude into the network.

f. Gaining control over the well-functioning network resources
As the firewall is absent in the network the intruders can easily intrude into the network and thereby the intruders can gain the access to well-functioning network devices

DOI 10.17148/IJARCCE.2016.51202

and network resources and thereby the intruder can simply make the network unavailable to authorized users and can stop intended network services.

g. Installation of rootkits
Intruders usually use rootkits to maintain their access to the victim's network thus they can install rootkits inside the network in order to maintain their access to network resources. This kind of activities can be blocked using a firewall. Thus the absence of firewall may help the intruder to install rootkits.

II. LITERATURE SURVEY

Mohamed M. Abd-Eldayem [4] has analyzed the Intrusion Detection System (IDM) based on Naı¨ve Bayes classifier. This technique identifies the most important HTTP traffic features that can be used to detect HTTP attacks. The main objective is to enhance IDS performance through preparing the training data set allowing detection of malicious connections that exploit the HTTP service. Annie shebanow, Richard Perez et.al. [5] explores how use, misuse, positive and negative, obstacle, and abuse testing cases of firewalls have broadened the security policies that mitigate or prevent threats in a cloud environment. According to them when comparing the efficacy of different types of firewalls, the focus should be on the capabilities of the data-transfer layers and the position of the firewalls. It is best to use the four layers of the Transmission Control Protocol/Internet Protocol (TCP/IP), which work together to transfer data between hosts: hardware (data), IP (network), transport, and the application layers. Chirag Modi, Dhiren Patel et.al. [6] has surveyed different intrusions techniques affecting availability, confidentiality and integrity of Cloud resources and services. It examines proposals incorporating Intrusion Detection Systems (IDS) in Cloud and discusses various types and techniques of IDS and Intrusion Prevention Systems (IPS), and recommends IDS/IPS positioning in Cloud architecture to achieve desired security in the next generation networks. Ehab Al-Shaer, Raouf boutaba et.al. [7] has identified different anomalies that could exist in a single- or multi-firewall environment. They also present a set of techniques and algorithms to automatically discover policy anomalies in centralized and distributed firewalls. Hazem Hamed, Ehab Al-Shaer [8] has presented a comprehensive classification of security policy conflicts that might potentially exist in a single security device (intra-policy conflicts) or between different network devices (inter-policy conflicts) in enterprise networks. Their approach is sufficiently general that it can be used to verify many other filtering based security policies such as authorization servers, intrusion detection, and intrusion prevention systems. Figure 1 shows the organization of their taxonomy for these conflicts.

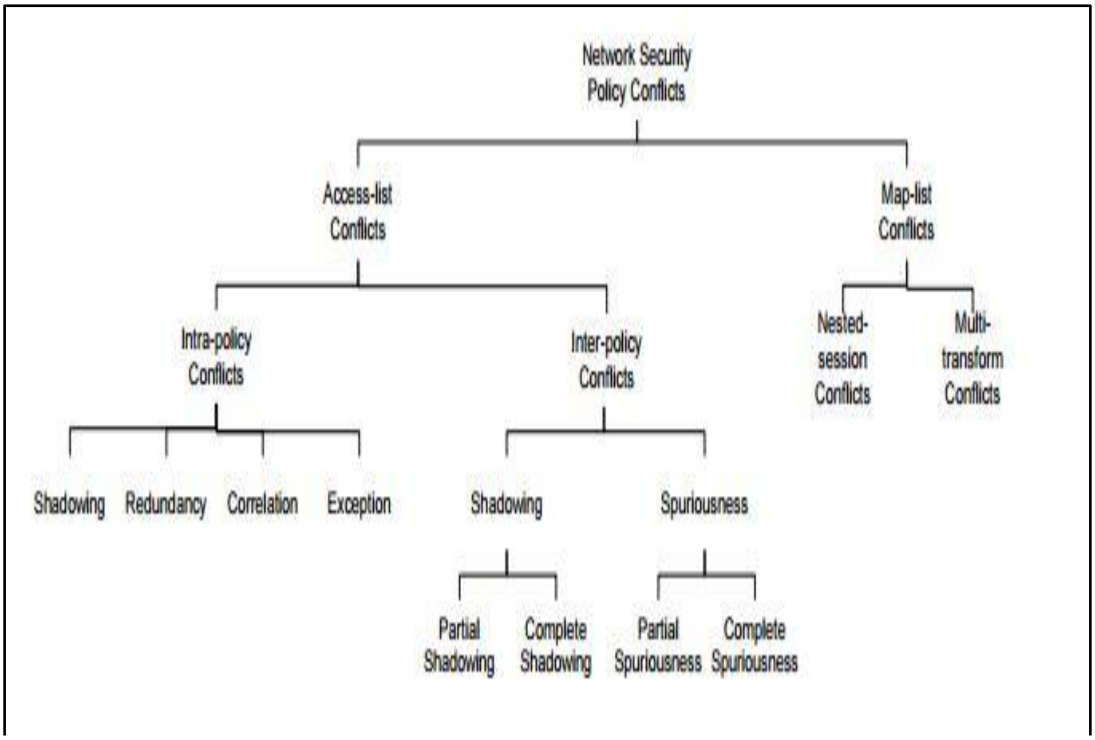

Figure 1 Classification chart of network security policy conflicts

Scott Hazlehurst [9] presented a new algorithm for representing the rules as a Binary Decision Diagram (BDD) and then shows how the resulting boolean expression can be used to analyze rule sets. According to him the efficacy of traffic management depends on how good these rules are. As per Scott Hazlehurst two kinds of problems exist in accessing and maintaining lists, they are:

DOI 10.17148/IJARCCE.2016.51202

- As the list of rules become more complex they become more difficult to understand. The person who maintains the list may leave and be replaced. Even changing the position of a rule in the rule list can change the semantics of the list.
- The cost of performing look-up on a rule list may become expensive, and particularly for routers, this may add significantly to latency in the network.

Seny Kamara, Sonia Fahmy et.al. [10] described a methodology for analyzing vulnerabilities in Internet firewalls. Firewall vulnerability is defined as an error made during firewall design, implementation, or configuration, which can be exploited to attack the trusted network that the firewall is supposed to protect. The result of their analysis is a set of matrices that illustrate the distribution of firewall vulnerability causes and effects over firewall operations. These matrices are useful in avoiding and detecting unforeseen problems during both firewall implementation and firewall testing. Liang Zhao, Akira Shimae, et.al. [11] has proposed a new yet simple technique called the linear-tree structure. It utilizes an advanced feature of modern firewalls, the “go to” like statement, to transform the given rule list into a rule set that is functionally equivalent to the original but organized in a more efficient structure. They stated it is possible to achieve much more improvement than previous, rule-reordering based studies. Ant Colony Optimization (ACO) was used to improve the performance of packet filter firewall rule configurations in [2]. ACO was also proved to be useful in feature optimization as well [12].

III.FIREWALL DESIGN AND IMPLEMENTATION

The following are the key constraints, which are to be kept in mind before designing a firewall

a. Goals
All incoming and outgoing traffic should pass through the firewall only. The traffic that satisfies with the local security policies should be allowed and the rest of the traffic should be denied by the firewall. Thus firewall should be placed at gateway position so that every data packet must pass through the firewall.

b. Selection criteria
The following are the few key criteria used to choose the best firewall for an organization.

I. Size of the organization
Before designing a firewall any organization or institution must consider the size of the organization. In terms or networking size of the organization refers to a total number of network attached devices.
II. Cost
Before purchasing any firewall (hardware or software firewall) it should be decided, whether the organization can afford for such firewall (i.e. the cost of the new firewall should be balanced with the advantages it provides).
III. Maintenance
The maintenance of the firewall plays a very vital role in network perimeter security. The firewall maintenance should be as easy as possible so that it can be completely used for what it is made
IV. Local resources and local security policy
Before purchasing or implementing a firewall it is mandatory to know what the firewall should protect, thus useful firewall can be implemented depending on our network resources and networking components. Any firewall can be considered as the useful firewall if it satisfies our local security policies and helps in making our security policies more efficient.
Netfilter is a set of hooks inside Linux kernel. It allows kernel modules to register call back functions with the network stack in order to intercept and manipulate the network packet.

IMPLEMENTATION

The Netfilter Hooks

Generally there exist 5 types of hooks in kernel modules, these Netfilter hooks are listed below along with their functioning. When a network packet comes in, it is passed to the Netfilter’s first hook NF_INET_PRE_ROUTING. After that, the packet goes through the routing code, which decides where the packet is destined to, either another port in same network interface or another interface. It also might drop the packet if it can’t be sent.
If the packet goes to another port in the same interface, the second hook NF_INET_LOCAL_IN is triggered. This happens before the packet reaches to the destination port.

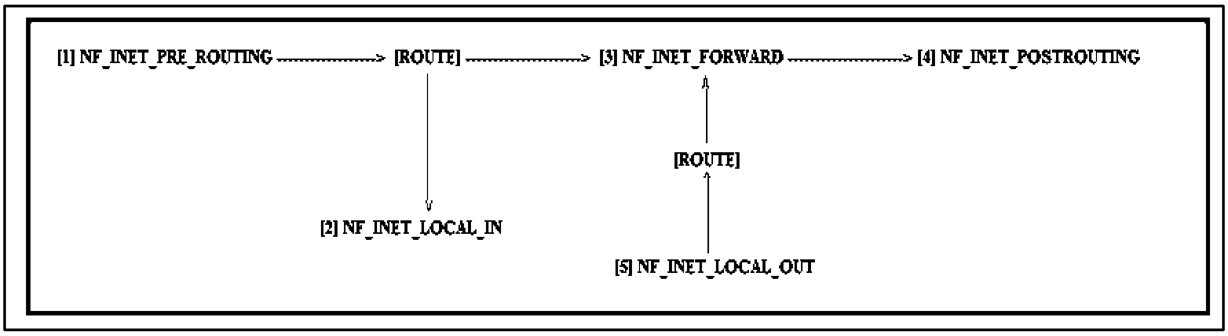

Figure 2 Netfilter System Hook

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 12, December 2016

If the packet goes to another network interface, the third hook NF_INET_FORWARD is called, followed by the fourth hook NF_INET_POST_ROUTING before the packet reaches wire again.

There is one hook left, NF_INET_LOCAL_OUT. It's called for local outgoing packets. Routing code is called before this hook to figure out the IP address and after this hook to decide the route.

Kernel modules can register to any of the 5 hooks.

The hook function prototype

Once a hook function has registered with any of the 5 Netfilter hooks, then it will be called when a network packet go through the hook in the network stack.

hooknum indicates one of the 5 hook types
NF_INET_PRE_ROUTING
NF_INET_LOCAL_IN
NF_INET_FORWARD
NF_INET_LOCAL_OUT
NF_INET_POST_ROUTING.

sbk is a pointer to the network packet buffer, which is defined in/lib/modules/$(uname-r)/build/include/Linux/skbuff.h.

in and **out** are two pointers to the net_device structure, which are what Linux kernel uses to describe network interface,as defined in /lib/modules/$(uname-r)/build/include/Linux/net device.h.

Therefore, depending on the packets traversal, either in or out will be NULL.

okfn is a function pointer enables registering of a callback function triggered when all the functions registered with this hook returned NF_ACCEPT, thus allow the packets.

When a registered function is called, it can do one of the five things and **return** the corresponding value, as defined in Netfilter.h.

NF_ACCEPT : let the packet pass.
NF_DROP : drop the packet.
NF_STOLEN : take the packet and don't let the packet pass.
NF_QUEUE : queue the packet, usually for userspace handling.
NF_REPEAT : call the hook again.

Access the Netfilter hooks

skb in the hook function enables one to access the network packet buffer and retrieve the needed information easily. Usually the three most important information fields, transport header, network_header and mac_header fields are defined in sk_buff.

One thing needs to mention is that skb_transport_header doesn't always work as expected. This function works when the network packets are processed by certain functions of Netfilter which is at the transport layer of Netfilter implementation.

When a packet goes out, it goes from the application layer, to transport layer, then network layer, and so on. These functions are normally executed and skb_transport_header works well. When a packet goes in from wire, it travels from the physical layer, data link layer, network layer and upwards, therefore it might not go through the functions defined in Netfilter for skb_transport_header to work. The skb_transport_header actually returns a pointer to the IP header, and the code skips the IP header by the length of typical IP header.

IV. RESULTS

PLUTO can run easily without any issues on a system with above configuration as it's already tested on the system configured with above-listed requirements.

PLUTO module info

```
root@mvgr-HP-PRO-330-MT:/nextgen/Desktop/pluto# modinfo pluto.ko
filename:       pluto.ko
author:         KRISHNA CHINTALAPUDI.
description:    pluto- A home office firewall
license:        GPL
srcversion:     4667253A4FCA1B47FA76ABD
depends:
vermagic:       2.6.35-22-generic SMP mod_unload modversions 686
```

Block all incoming traffic, unblock all outgoing traffic

Enter the configuration commands below

```
root@mvgr-HP-PRO-330-MT:/nextgen/Desktop/pluto# ./usm --in --proto ALL --action BLOCK
root@mvgr-HP-PRO-330-MT:/nextgen/Desktop/pluto# ./usm --out --proto ALL --action UNBLOCK
root@mvgr-HP-PRO-330-MT:/nextgen/Desktop/pluto# ./usm –print
```

in/out	src ip	src mask	src port	dest ip	dest mask	dest port	proto	action
in	-	-	-	-	-	-	ALL	BLOCK
out	-	-	-	-	-	-	ALL	UNBLOCK

Test IP address and Netmask

```
root@mvgr-HP-PRO-330-MT:/nextgen/Desktop/pluto# ./usm --out --srcip 192.168.5.25 --proto UDP --action BLOCK
root@mvgr-HP-PRO-330-MT:/nextgen/Desktop/pluto# ./usm –print
```

in/out	src ip	src mask	src port	dest ip	dest mask	dest port	proto	action
out	192.168.5.25	-	-	-	-	-	UD	BLOCK

```
root@mvgr-HP-PRO-330-MT:/nextgen/Desktop/pluto# ping google.com
ping: unknown host google.com
root@mvgr-HP-PRO-330-MT:/nextgen/Desktop/pluto# ./usm --delete 1
root@mvgr-HP-PRO-330-MT:/nextgen/Desktop/pluto# ./usm --out --srcip 192.168.5.116 --proto UDP --action BLOCK
root@mvgr-HP-PRO-330-MT:/nextgen/Desktop/pluto# ./usm –print
```

```
in/out   src ip          src mask   src port   dest ip   dest mask   dest port   proto   action
out      192.168.5.116   -          -          -         -           -           UDP     BLOCK
```

V. CONCLUSION

In this paper, we have examined firewall design thoroughly in an attempt to meet security and performance requirements of multitier applications. In all scenarios, servers hosting application components were separated from the company's corporate network used to conduct internal business, as an initial step to segregate resources with different security requirements. To tightly control interactions between the application's tiers, we looked at hosting tiers of the application on dedicated subnets. By deploying “PLUTO – A home office firewall”, we were able to significantly increase the difficulty of obtaining unauthorized access to sensitive resources from the Internet. At the same time, our firewall decreases the contributing to the cost of deploying and maintaining the infrastructure, and decreasing the likelihood that it will be misconfigured.

REFERENCES

[1] Ravi Kiran Varma P, Valli Kumari V, and Srinivas Kumar S, "Feature selection using relative fuzzy entropy and ant colony optimization applied to real-time intrusion detection system," Procedia Computer Science, vol. 85, no. 2016, pp. 503-510, 2016.

[2] Ravi Kiran Varma P, Valli Kumari V, and Srinivas Kumar S, "Ant Colony Optimization-based Firewall Anomaly Mitigation Engine," Springerplus, vol. 5, no. 1, pp. 1-32, 2016.

[3] Free Computer tutorials. [Online]. http://www.homeandlearn.co.uk/BC//bcs5p4.html

[4] Mohamed M. Abd-Eldayem, "A proposed HTTP service based IDS," Egyptian informatics journal, no. 15, pp. 13-24, January 2014.

[5] Annie Shebanow, Richard Perez, and Caroline Howard, "The effect of firewall testing types on cloud security policies," International Journal of Strategic Information Technology and Applications, pp. 60-68, September 2012.

[6] Chirag Modi et al., "A survey of intrusion detection techniques in the cloud," Journal of Network and Computer Applications, vol. 36, no. 1, pp. 42-57, 2013.

[7] Ehab Al-Shaer, Hazem Hamed, Raouf Boutaba, and and Masum Hasan, "Conflict Classification and Analysis of Distributed Firewall Policies," IEEE journal on selected areas in communication, vol. 25, no. 10, October 2005.

[8] Hazem Hamed and Ehab Al-Shaer, "Taxonomy of conflicts in network security policies," IEEE Communications Magazine, vol. 44, no. 3, pp. 134-141, March 2006.

[9] Scott Hazlehurst. (2000, August) Algorithms for Analysing Firewall and Router Access Lists.

[10] Seny Kamara, Sonia Fahmy, Eugene Schultz, Florian Kerschbaum, and Michael Frantzen. Analysis of Vulnerabilities in Internet Firewalls. Center for Education and Research in Information Assurance and Security (CERIAS),Purdue University.

[11] Akira Shimae, Hiroshi Nagamochi. Liang Zhao, "Linear-tree rule structure for firewall optimization," in The Sixth IASTED International Conference on Communications, Internet, and Information Technology, Banff, Alberta, Canada, 2007, pp. 67-72.

[12] Ravi Kiran Varma P, Valli Kumari V, and Srinivas Kumar S, "A Novel Rough Set Attribute Reduction based on Rough Sets and Ant Colony Optimization," International Journal Intelligent Systems Technologies and Applications, vol. 14, no. 3/4, pp. 330-353, 2015.