ISSN: 1991-8941

AN IMPLEMENTATION OF FIREWALL SYSTEM USING MIKROTIK ROUTER OS

Shaymaa W. Abdulatteef
Al-Nahrain University - Computer Engineering Department .

Abstract- This work concerned with implementing packet filtering firewall by using Mikrotik Router OS, and tested on a LAN. In the host machine, the programs VMware workstation and Wireshark were installed. Mikrotik Router OS give the same results of real environment. This work aims to drop unwanted packets according to rules defined to Mikrotik Router OS including source address, destination address, source & destination port and specified action written in command line window.

Keywords: IMPLEMENTATION , FIREWALL , MIKROTIK ROUTER OS

Introduction

The increasing complexity of networks, and the need to make them more open due to the growing emphasis on and attractiveness of the internet as a medium for business transactions, mean that networks are becoming more and more exposed to attacks. Once attached to the internet, in addition to taking advantage of its many benefits without risks, the ‘connected organization’ needs to protect it from attack. The search is on for mechanisms and techniques for the protection of internal networks from such attacks. One of the protective mechanisms under serious consideration is the firewall. A firewall protects a network by guarding the points of entry to it. Firewalls are becoming more sophisticated by the day, and new features are constantly being added.

The firewall selectively controls the flow of data to and from network. Packet filters allow or block packets, usually while routing from one network to another (most often from the Internet to an internal network, and vice versa). To accomplish packet filtering, a set of rules must set up, that specify what types of packets e.g., those to or from a particular IP address or port are be allowed, other types are be blocked. Packet filtering may occur in a router, in a bridge, or on an individual host. It is sometimes known as screening [1]. The fundamental function of a firewall is to restrict the flow of information between two networks. To set a firewall, the administrator should define what kinds of data pass and what kinds are blocked. This is called defining of the firewall's policy. The policy instruct the firewall as to how it should control the traffic that traverse between internal trusted networks and external untrusted and unknown networks . Two default policies are possible, the first is default forward (allow list): which include the rules to allow authorized users to freely pass through the firewall. In other words, that which is not expressly prohibited is permitted. And the second is default discard (disallow list): which include the rules to keep unauthorized users from gaining access to an internal network. In other words, that which is not expressly permitted is prohibited [2, 3].

System Design

The mechanism used is Packet filtering, as shown in figure 1 the implemented system depends on the header attributes which is the interest for packet filtering (i.e. the filtering operation depends on checking of these header attributes).

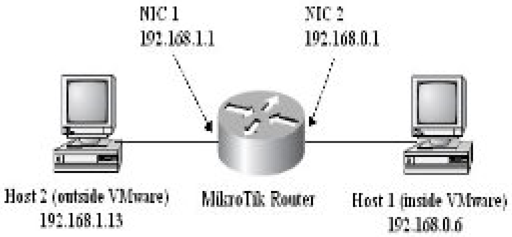


Figure 1. packet filtering design

The main processes done by the system described flowchart shown in figure 2; the system will be described in general.

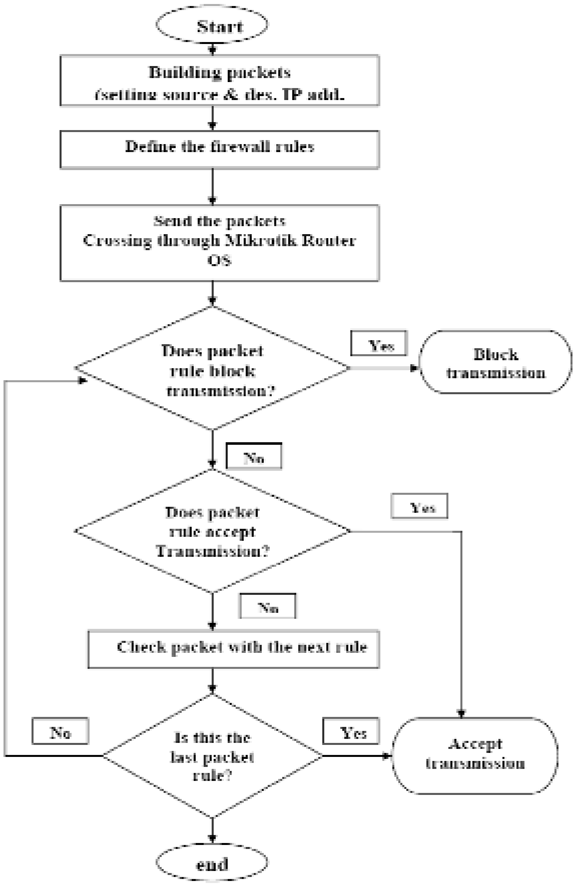


Figure 2. System Flowchart

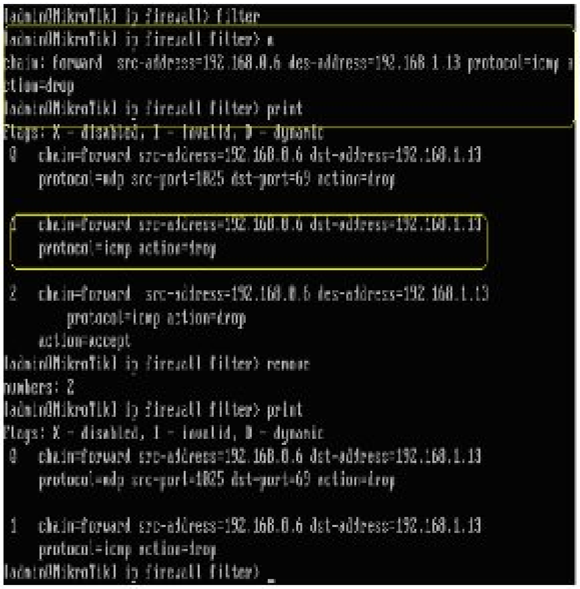
SYSTEM IMPLEMENTATION

The system tested to filter packets on three types of protocols which are:

1. ICMP scenario (reply echo message)

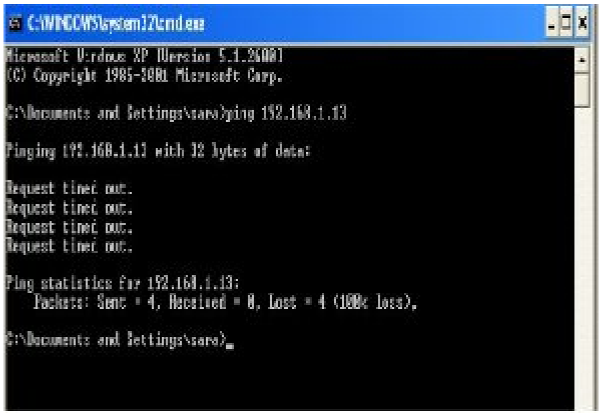
The packet has been built with determining IP packet, source address and destination address, and then the rule chain was built with in Mikrotik Router through IP firewall filter add command chain based on forward packet with src-add, des-add, protocol=icmp, and action=drop. as shown in Figure 3, Figure 4 describes the dropping of an ICMP packet, then a rule has been built to accept the packet

>ip firewall filter add chain: forward src-address=192.168.0.6 des-address =192.168.1.13 protocol =icmp action=accept as shown in Figure 5, Figure 6 describes the accepting of an ICMP packet



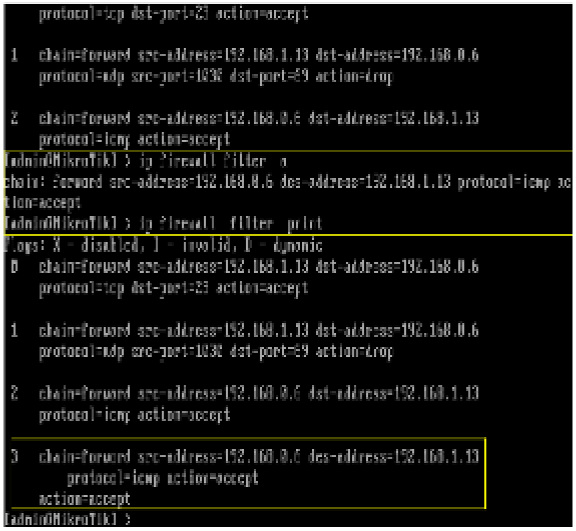
```
[admin@MikroTik] ip firewall> filter
[admin@MikroTik] ip firewall filter> a
chain: forward  src-address=192.168.0.6 des-address=192.168 1.13 protocol=icmp a
ction=drop
[admin@MikroTik] ip firewall filter> print
Flags: X - disabled, I - invalid, D - dynamic
 0   chain=forward src-address=192.168.0.6 dst-address=192.168.1.13
     protocol=udp src-port=1025 dst-port=69 action=drop

 1   chain=forward src-address=192.168.0.6 dst-address=192.168.1.13
     protocol=icmp action=drop

 2   chain=forward  src-address=192.168.0.6 des-address=192 168.1.13
        protocol=icmp action=drop
     action=accept
[admin@MikroTik] ip firewall filter> remove
numbers: 2
[admin@MikroTik] ip firewall filter> print
Flags: X - disabled, I - invalid, D - dynamic
 0   chain=forward src-address=192.168.0.6 dst-address=192.168.1.13
     protocol=udp src-port=1025 dst-port=69 action=drop

 1   chain=forward src-address=192.168.0.6 dst-address=192.168.1.13
     protocol=icmp action=drop
[admin@MikroTik] ip firewall filter> _
```

Figure 3. Defining ICMP rule with drop action



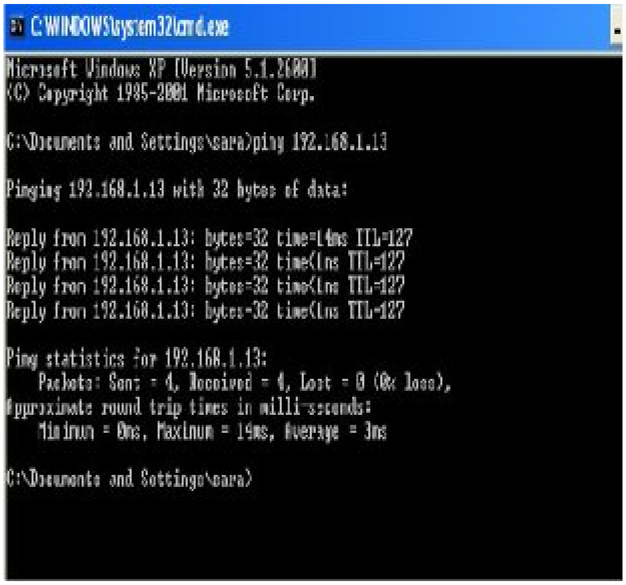
```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\sara>ping 192.168.1.13

Pinging 192.168.1.13 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.1.13:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\Documents and Settings\sara>_
```

Figure 4. Describes the dropping of an ICMP packet



```
     protocol=tcp dst-port=23 action=accept

 1   chain=forward src-address=192.168.1.13 dst-address=192.168.0.6
     protocol=udp src-port=1030 dst-port=69 action=drop

 2   chain=forward src-address=192.168.0.6 dst-address=192.168.1.13
     protocol=icmp action=accept
[admin@MikroTik] > ip firewall filter a
chain: forward src-address=192.168.0.6 des-address=192.168.1.13 protocol=icmp ac
tion=accept
[admin@MikroTik] > ip firewall  filter  print
Flags: X - disabled, I - invalid, D - dynamic
 0   chain=forward src-address=192.168.1.13 dst-address=192.168.0.6
     protocol=tcp dst-port=23 action=accept

 1   chain=forward src-address=192.168.1.13 dst-address=192.168.0.6
     protocol=udp src-port=1030 dst-port=69 action=drop

 2   chain=forward src-address=192.168.0.6 dst-address=192.168.1.13
     protocol=icmp action=accept

 3   chain=forward src-address=192.168.0.6 des-address=192.168.1.13
        protocol=icmp action=accept
     action=accept
[admin@MikroTik] >
```

Figure 5. Defining ICMP rule with accepted action



```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\sara>ping 192.168.1.13

Pinging 192.168.1.13 with 32 bytes of data:

Reply from 192.168.1.13: bytes=32 time=14ms TTL=127
Reply from 192.168.1.13: bytes=32 time<1ms TTL=127
Reply from 192.168.1.13: bytes=32 time<1ms TTL=127
Reply from 192.168.1.13: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.1.13:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 14ms, Average = 3ms

C:\Documents and Settings\sara>
```

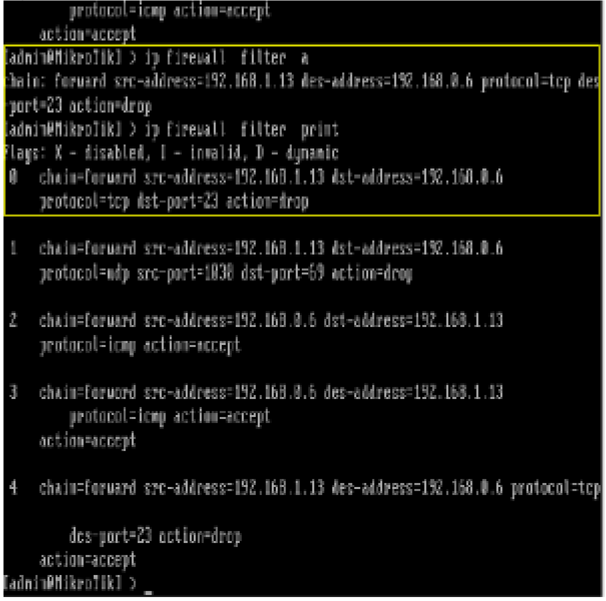
Figure 6. Accepting an ICMP packet

Telnet as a port of TCP protocol.

The telnet is enabled on the 192.168.0.6(virtual pc), the rule is built to drop remote access from 192.168.1.13, src-port number=23 as shown in figure 7.

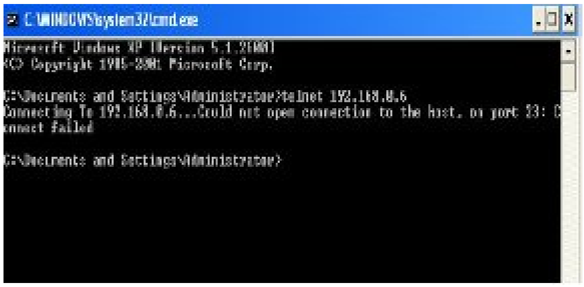
>ip firewall filter add chain src=192.168.0.6 des=192.168.1.13 protocol=tcp des-port=23 action=drop

Then from 192.168.1.13(real pc) telnet to 192.168.0.6(virtual pc) is not allowed as shown in figure 8.



```
        protocol=icmp action=accept
    action=accept
[admin@MikroTik] > ip firewall  filter  a
chain: forward src-address=192.168.1.13 des-address=192.168.0.6 protocol=tcp des
-port=23 action=drop
[admin@MikroTik] > ip firewall  filter  print
Flags: X - disabled, I - invalid, D - dynamic
 0   chain=forward src-address=192.168.1.13 dst-address=192.168.0.6
     protocol=tcp dst-port=23 action=drop

 1   chain=forward src-address=192.168.1.13 dst-address=192.168.0.6
     protocol=udp src-port=1030 dst-port=69 action=drop

 2   chain=forward src-address=192.168.0.6 dst-address=192.168.1.13
     protocol=icmp action=accept

 3   chain=forward src-address=192.168.0.6 des-address=192.168.1.13
        protocol=icmp action=accept
     action=accept

 4   chain=forward src-address=192.168.1.13 des-address=192.168.0.6 protocol=tcp

        des-port=23 action=drop
     action=accept
[admin@MikroTik] > _
```

Figure 7. Defining TCP rule for drop telnet access

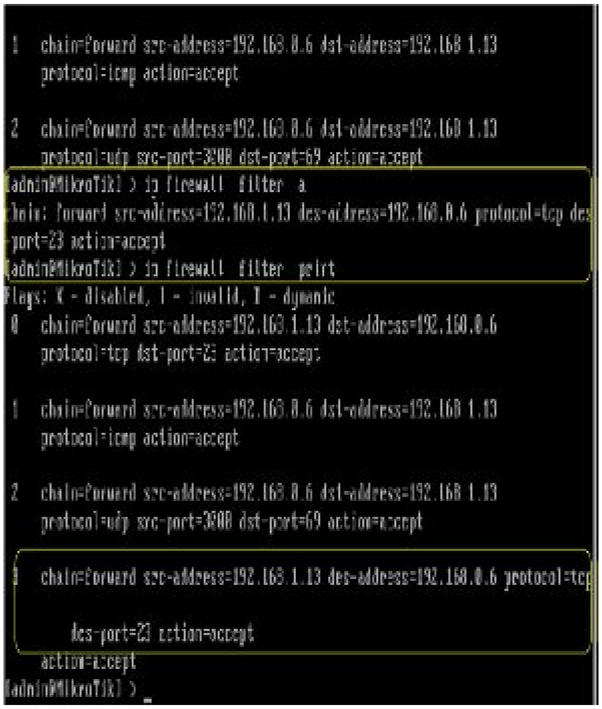


```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Administrator>telnet 192.168.0.6
Connecting To 192.168.0.6...Could not open connection to the host, on port 23: C
onnect failed

C:\Documents and Settings\Administrator>
```

Figure 8. Describes the Fail connection on telnet port

Then a new rule is built to accept login remotely into pc1(virtual pc) via telnet service as shown in figure 9.

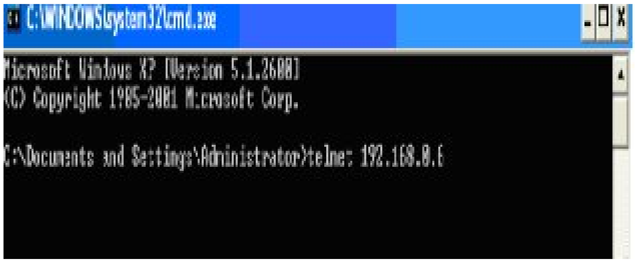
>ip firewall filter add chain src=192.168.0.6 des=192.168.1.13 des-port=23 action accept



```
 1   chain=forward src-address=192.168.0.6 dst-address=192.168.1.13
     protocol=icmp action=accept

 2   chain=forward src-address=192.168.0.6 dst-address=192.168.1.13
     protocol=udp src-port=3000 dst-port=69 action=accept
[admin@MikroTik] > ip firewall  filter  a
chain: forward src-address=192.168.1.13 des-address=192.168.0.6 protocol=tcp des
-port=23 action=accept
[admin@MikroTik] > ip firewall  filter  print
Flags: X - disabled, I - invalid, D - dynamic
 0   chain=forward src-address=192.168.1.13 dst-address=192.168.0.6
     protocol=tcp dst-port=23 action=accept

 1   chain=forward src-address=192.168.0.6 dst-address=192.168.1.13
     protocol=icmp action=accept

 2   chain=forward src-address=192.168.0.6 dst-address=192.168.1.13
     protocol=udp src-port=3000 dst-port=69 action=accept

 3   chain=forward src-address=192.168.1.13 des-address=192.168.0.6 protocol=tcp

        des-port=23 action=accept
     action=accept
[admin@MikroTik] > _
```

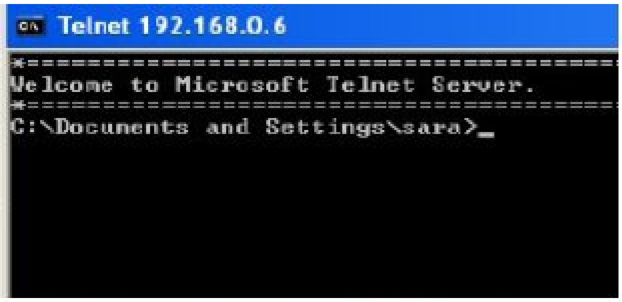
Figure 9. Defining rule for accept TCP connection on telnet port

And pc2(192.168.1.13) could login remotely into pc1 (192.168.0.6) according to the obvious rule, there are the steps of accept login as shown in figures 10 and 11.



```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Administrator>telnet 192.168.0.6
```

Figure 10. Examine rule by telnet IP address

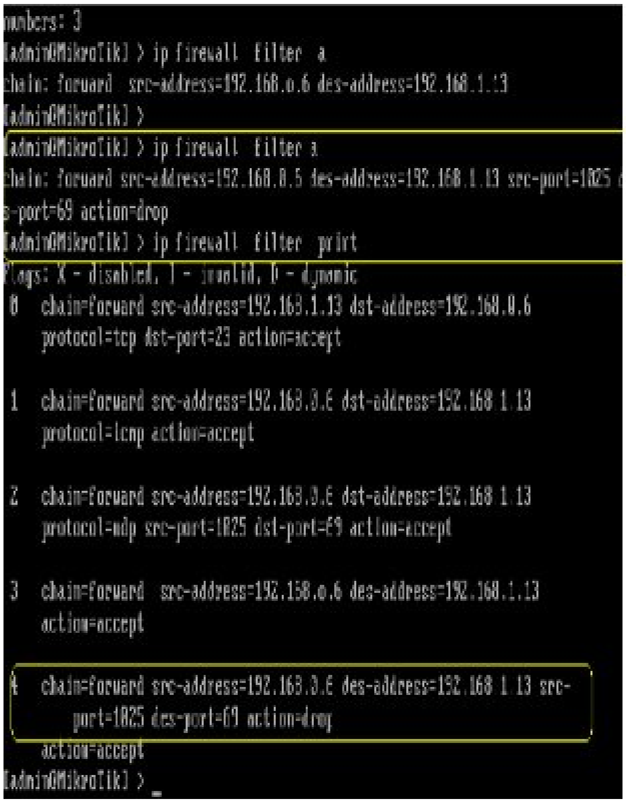


```
Telnet 192.168.0.6
*===============================================
Welcome to Microsoft Telnet Server.
*===============================================
C:\Documents and Settings\sara>_
```

Figure11. Describes the successful telnet login

Tested the firewall on a specified UDP packet.

First rule chain is built and defined to MIKROTIK ROUTER to drop the packet as shown in figure 12.
>ip firewall filter add chain src=192.168.0.6 des=192.168.1.13 src-port=1025 des-port=69 action drop



```
numbers: 3
[admin@MikroTik] > ip firewall  filter  a
chain: forward  src-address=192.168.o.6 des-address=192.168.1.13
[admin@MikroTik] >
[admin@MikroTik] > ip firewall  filter a
chain: forward src-address=192.168.0.5 des-address=192.168.1.13 src-port=1025 d
s-port=69 action=drop
[admin@MikroTik] > ip firewall  filter  print
Flags: X - disabled, I - invalid, D - dynamic
 0   chain=forward src-address=192.168.1.13 dst-address=192.168.0.6
     protocol=tcp dst-port=23 action=accept

 1   chain=forward src-address=192.168.0.6 dst-address=192.168.1.13
     protocol=icmp action=accept

 2   chain=forward src-address=192.168.0.6 dst-address=192.168.1.13
     protocol=udp src-port=1025 dst-port=69 action=accept

 3   chain=forward  src-address=192.158.o.6 des-address=192.168.1.13
     action=accept

 4   chain=forward src-address=192.168.0.6 des-address=192.168.1.13 src-
        port=1025 des-port=69 action=drop
     action=accept
[admin@MikroTik] > _
```

Figure 12. Defining UDP rule with drop action

Then the packet sent through Wireshark whose capture any packet coming to 192.168.1.13, the packet is blocked as shown in figure 13

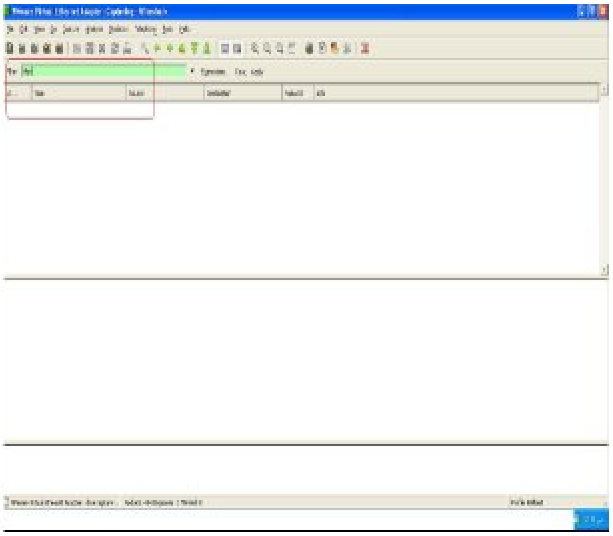
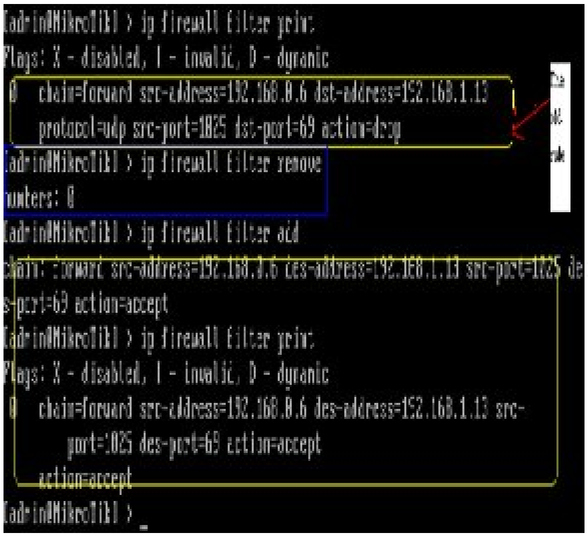


Figure 13. Shows that Wireshark couldn't capture UDP packet

Then a new rule is built to accept the packet as shown in figure 14.
>ip firewall filter add chain src=192.168.0.6 des=192.168.1.13 src-port=1025 des-port=23 action accept



```
[admin@MikroTik] > ip firewall filter print
Flags: X - disabled, I - invalid, D - dynamic
 0   chain=forward src-address=192.168.0.6 dst-address=192.168.1.13
     protocol=udp src-port=1025 dst-port=69 action=drop
[admin@MikroTik] > ip firewall filter remove
numbers: 0
[admin@MikroTik] > ip firewall filter add
chain: forward src-address=192.168.0.6 des-address=192.168.1.13 src-port=1025 de
s-port=69 action=accept
[admin@MikroTik] > ip firewall filter print
Flags: X - disabled, I - invalid, D - dynamic
 0   chain=forward src-address=192.168.0.6 des-address=192.168.1.13 src-
        port=1025 des-port=69 action=accept
     action=accept
[admin@MikroTik] > _
```

Figure 14. Remove (drop UDP packet rule) and define (accept UDP packet rule)

Wireshark monitor and accepting packet is captured as shown in figure 15.

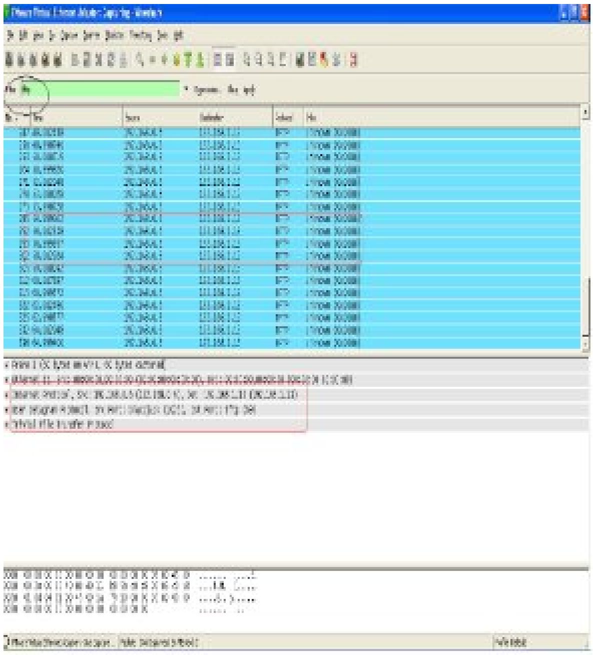


Figure 15. Shows Wireshark capture accepted UDP packet

Conclusions

In this paper I conclude that this design provided a relatively low cost, high reliability solution, since the fastest and the cheapest one of firewall techniques is packet filtering and since MIKROTIK ROUTER OS gave the same results of real environment.

I suggest implementing the firewall system using another type of its technologies as a future work.

References

[1] T.Ogletree, (2000). Practical Firewalls. First Edition. Que.USA.

[2] M.Strebe, C.Perkins, (2002) .Firewall 24Seven. 2nd Edition. Sybex Puplishing.

[3] E.D.Zwicky, S.Cooper, (2000).Building Internet Firewalls. 2nd Edition. O'Reilly Media. USA.

[4] MikroTik RouterOS ™ v2.9, Reference Manual http://www.mikrotik.com

تنفيذ نظام الجدار الناري بأستخدام جهاز التوجيه MIKROTIK OS

شيماء وليد عبد اللطيف

E.mail: mortadha61@yahoo.com

الخلاصة:

هذا العمل يهتم بتنفيذ جدار حماية تصفية الحزمة باستخدام جهاز التوجيه MIKROTIK OS, واختبارها على شبكة منطقة محلية. في الجهاز المضيف , تم تثبيت البرامج VMware workstation و Wireshark. جهاز التوجيه MIKROTIK OS يعطي نفس النتائج في البيئة الحقيقية. ويهدف هذا العمل الى اسقاط الحزم غير المرغوب فيها وفقآ لقواعد محددة في جهلز التوجيه MIKROTIK OS بما في ذلك عنوان المصدر وعنوان الوجهة, ونفذ المصدر والوجهه واجراءات محددة في نافذة كتابة سطر الاوامر .