Taking advantage of MikroTik RouterOS inbuilt firewall

by Martin Pína

MUM Croatia, Zagreb, 15th March, 2013

my background

- network and Linux administrator
- security consultant
- tailored services
- Certifications
  - Cisco Certified Academy Instructor
  - Netasq Certified Expert Plus
  - MikroTik Certified Trainer up to MTCINE

MikroTik Firewall

- IPv4 and IPv6
- same concept as Linux iptables
- up to L7
- L2 bridges offer additional filters

Goal

- fully automated and predefined, after initial configuration, no or minimal maintenance required, maintenance scripts
- designed for dynamic changes
- provides protection for customers/servers
- can detect and possibly stop suspicious traffic
- can be monitored

Other possibilities

- passive detection
  - monitoring
  - alerting
- dedicated IPS/IDS
  - Open source – SNORT®
  - Commercial – NETASQ

Advantages and disadvantages of special IDS/IPS

⊕ more possibilities

⊕ offload router's processing power

⊖ dedicated box

⊖ usually more expensive

note: may be achieved by virtual server inside MikroTik RouterOS

Deployment considerations

- ethics of network neutrality and different attitude for
  - corporate networks
  - ISP networks
  - hotspots
- position in network
- choosing appropriate action
  - drop
  - reject

Common „outside“ threats

- DoS/DDoS (usually SYN flood)
- bandwidth attack
- brute force for access
- bogons

Common „inside“ threats

- infected computers
- spammers
- corporate network leaking through unauthorized VPNs
- rogue or misconfigured DHCP server
- infrastructure attacks OSPF/STP/VRRP
- IP spoofing and ARP spoofing

What to protect?

- end hosts
  - human operated computers
  - servers
  - special purpose network devices
- network infrastructure
  - routers
  - switches

RouterOS protection

- management
  - SSH
  - winbox
  - ...
- network services
  - DHCP
  - webproxy
  - BGP
  - ...

How?

- disable unnecessary services

  e.g.: /ip service disable [/ip service find where name=telnet]

- control MAC-telnet and btest server

  e.g.: /tool mac-server set interface=ether10 disabled=yes

- Firewall

Why Firewall?

⊕ it is there for exactly the purpose

⊕ address-list feature enables you to work with „objects“ rather than addresses

⊕ allows creating dynamic address lists

⊕ it can be used even for bridged communication with use of

/interface bridge settings set use-ip-firewall=yes

⊕ is easily scriptable

Why not

- ⊖ extra work
- ⊖ may be CPU demanding
- ⊖ is not good for everything
- ⊖ may be nightmare if making false positives

Whitelisting management access

- create an address-list entry

  e.g.:
  ```
  /ip firewall address-list add list=ipv4-management-access address=192.168.0.60/32 disabled=no
  ```

- allow management only for addresses from address list

  e.g.:
  ```
  /ip firewall filter add chain=input protocol=tcp dst-port=22 src-address-list=!ipv4-management-access action=drop
  ```

ICMP/Ping floods

- Allow only reasonable amount of ICMP going through

  ```
  /ip firewall filter add chain=forward
  protocol=icmp limit=5/1,10 action=accept
  ```

  ```
  /ip firewall filter add chain=forward
  protocol=icmp action=drop
  ```

- Can be customized for
  - specific ICMP type/code
  - different chains

SYN floods #1

- limit

  ```
  /ip firewall filter add chain=input protocol=tcp dst-port=80 connection-limit=10,32 action=add-src-to-address-list address-list=ipv4-input-port80-over-limit address-list-timeout=1d
  ```

  ```
  /ip firewall filter add chain=input src-address-list=ipv4-input-port80-over-limit action=drop
  ```

- be careful about the limits and specify the rule as much as possible

SYN floods #2

- tarpit

  slows down the incoming connections

  e.g.: /ip firewall filter add chain=input protocol=tcp tcp-flags=syn connection-state=new dst-port=80 connection-limit=10,32 src-address-list=ipv4-input-port80-over-limit action=tarpit
- be carefull about „limit“ and „connection-limit“

rogue DHCP servers

- detect rogue DHCP servers

  /ip firewall filter add chain=input protocol=udp src-port=67 in-bridge-port=ether1 action=drop

- be carefull about input and forward chain
- you can optionally redirect the client's http traffic to informative web page
- There is also /ip dhcp-server alert option which can use script to add the rogue dhcp server's address to firewall

IP spoofing

- accept only src-addresses from given network
- can be only used on end-routers subnet
- in RouterOS v6, there is rp-filter option
- you can script it easily

```
:local interfacesubnet [/ip address get [/ip address find where interface=ether2] address]; /ip firewall address-list add address=$interfacesubnet list=ipv4-ether1-interfacesubnet
```

ARP spoofing

- static ARP or ARP reply-only can be used, but it is lacking detection and alerting
- there are endless possibilities using bridge filter

```
/interface bridge filter add arp-

arp-dst-address  arp-dst-mac-address  arp-gratuitous  arp-hardware-type  arp-opcode  arp-packet-type  arp-src-address  arp-src-mac-address
```

Port scanning #1

- there is nice inbuilt feature of firewall to detect and sort port scanning

  e.g.:

  ```
  /ip firewall add chain=input protocol=tcp psd=30,3s,3,1 action=add-src-to-address-list address-list="ipv4-portscan" address-list-timeout=1w
  ```

Port scanning #2

- You can also detect specific types of scans

  e.g.: null scan

  ```
  /ip firewall filter add chain=input protocol=tcp tcp-flags=!fin,!syn,!rst,!psh,!ack,!urg action=add-src-to-address-list address-list="ipv4-portscan" address-list-timeout=1w
  ```

- After detection it is needed to treat the IPs from portscan address-list

L7 rules

- may be used as a protection for keeping port numbers used for their intended purpose
- may be used as a protection for forged packet attacks to weak network services protocols implementations

Greylisting

- address lists may be used recursively with to effectively greylist some addresses for defined time
- this method is also used for port knocking feature which may be used as intended backdoor to the router in case the rules are so strict that administrator cannot log in

Monitoring

- if unsure action=passtrough may be used just to see the counters
- if some security issues are considered critical, it can be logged by action=log and/or processed further by scripting

Proposals

- use address lists (with timeouts)
- use structured firewall
- use comments
- automate everything

Caveat

- CPU (L7!)
- FastPath
- False positives

Resources

Wiki

- http://wiki.mikrotik.com/wiki/Manual:IP/Firewall
- http://wiki.mikrotik.com/wiki/Firewall

Presentations

- Tom Smyth's
- Wardner Maia

Thanks for your attention (patience)

- Questions... ?

hanácké.net®