NetFilter – IPtables

- Firewall
  - Series of rules to govern what Kind of access to allow on your system
  - Packet filtering
  - Drop or Accept packets
- NAT
  - Network Address Translation
- Modularized -- Modules loaded as part of service

Netfilter Web Site

- www.netfilter.org

- http://www.netfilter.org/documentation/HOWTO//packet-filtering-HOWTO.html

Linux 2.4 Packet Filtering HOWTO
Rusty Russell, mailing list netfilter@lists.samba.org
$Revision: 1.26 $ $Date: 2002/01/24 13:42:53 $

Were on you System is it?

- /etc/sysconfig/iptables
- /etc/sysconfig/iptables.save
- /etc/sysconfig/iptables-config
- /etc/rc.d/init.d/iptables
- system-config-secuitylevel

- Ref: Page 434

iptables Sevice Script

- Service command does not start or stop iptables service it act as a management tool
- service iptables stautus
  - list current rules
- service iptables stop
  - flushes current rules
- service iptables start
  - flushes current rules and adds from iptables file
- service iptables save
  - saves current rules to iptables file

Netfilter – Packet Filtering

- Framework for packet management
- Checks packets for network protocols and notifies parts of kernel listening for them
- IPtables is built on this framwork
- Netfilter supports three tables:
  - filter, nat, and mangle
- Packet filter is implemented using a filter table that holds rules for dropping or accepting packets
- NAT table holds rule for address translation such as masquerading
- Mangle table is used for specialized packet changes

Chains

A chain is simply a check list of rules. These rules specify what action to take for packets containing certain headers. If the target does match a rule it is passed on to the target. If a packet does not match the first rule the next rule is checked. If the packet does not mach any rules, the kernel checks the chain policy. Usually the packet is dropped or rejected

Chain names have to be entered in upper case.

- INPUT
- OUTPUT
- FORWARD
- PREROUTING
- POSTROUTING

REF: Pages 422

Targets

There are two built in targets DROP and ACCEPT. Other targets can be user defined chains or extension add on such as REJECT.

- ACCEPT
- DROP
- REJECT
- QUEUE
- RETURN

REF: Page 423

iptables Command

- Manage IP table rules
- Table must be specify if not the default filter table i.e.: iptables –t nat
- iptables –L to list active rules
- iptables –A *chain* to add rule
- iptables –D *chain* to delete rule
- ! symbol turns a rule into its inverse

Option	Function
`-A` *chain*	Appends a rule to a chain.
`-D` *chain* [*rulenum*]	Deletes matching rules from a chain. Deletes rule *rulenum* (1 = first) from *chain.*
`-I` *chain* [*rulenum*]	Inserts in *chain* as *rulenum* (default 1 = first).
`-R` *chain rulenum*	Replaces rule *rulenum* (1 = first) in *chain.*
`-L` [*chain*]	Lists the rules in *chain* or all chains.
`-E` [*chain*]	Renames a chain.
`-F` [*chain*]	Deletes (flushes) all rules in *chain* or all chains.
`-R` *chain*	Replaces a rule; rules are numbered from 1.
`-Z` [*chain*]	Zero counters in *chain* or all chains.
`-N` *chain*	Creates a new user-defined chain.
`-X` *chain*	Deletes a user-defined chain.
`-P` *chain target*	Changes policy on *chain* to *target.*

Option	Function
`-p [!]` *proto*	Specifies a protocol, such as TCP, UDP, ICMP, or ALL.
`-s [!]` *address* [*/mask*] `[!]` [*port* [*:port*]]	Source address to match. With the *port* argument, you can specify the port.
`--sport [!] [`*port* [*:port*]]	Source port specification. You can specify a range of ports using the colon, *port:port.*
`-d [!]` *address* [*/mask*] `[!]` [*port* [*:port*]]	Destination address to match. With the *port* argument, you can specify the port.
`--dport [!]` [*port* [*:port*]]	Destination port specification.
`--icmp-type [!]` *typename*	Specifies ICMP type.
`-i [!]` *name* `[+]`	Specifies an input network interface using its name (for example, **`eth0`**). The `+` symbol functions as a wildcard. The `+` attached to the end of the name matches all interfaces with that prefix (**`eth+`** matches all Ethernet interfaces). Can be used only with the INPUT chain.
`-j` *target* **`[port]`**	Specifies the target for a rule (specify **`[port]`** for REDIRECT target).
`--to-source` < *ipaddr*> [-< *ipaddr*>] [*: port- port*]	Used with the SNAT target, rewrites packets with new source IP address.

Option	Function
`--to-destination` <*ipaddr*> [-< *ipaddr*>] [*: port- port*]	Used with the DNAT target, rewrites packets with new destination IP address.
`-n`	Numeric output of addresses and ports, used with `-L`.
`-o [!]` *name* `[+]`	Specifies an output network interface using its name (for example, `eth0`). Can be used only with FORWARD and OUTPUT chains.
`-t` *table*	Specifies a table to use, as in `-t nat` for the NAT table.
`-v`	Verbose mode, shows rule details, used with `-L`.
`-x`	Expands numbers (displays exact values), used with `-L`.
`[!] -f`	Matches second through last fragments of a fragmented packet.
`[!] -V`	Prints package version.
`!`	Negates an option or address.
`-m`	Specifies a module to use, such as state.
`--state`	Specifies options for the state module such as NEW, INVALID, RELATED, and ESTABLISHED. Used to detect packet's state. NEW references SYN packets (new connections).
`--syn`	SYN packets, new connections.

Examples

- iptables –A INPUT -s 10.161.24.0/23 –j ACCEPT
- iptables –N incoming
  - User defined chain
- iptables –A incoming –j DROP -i eth0 –s 10.161.24.1
- iptables –A incoming –j ACCEPT –i lo
  - Denies traffic from source 10.161.24.1 and allows from localhost
- iptables –A INPUT –j incoming
- iptaples –A FORWARD –j incoming
  - points target to user defined chain
- iptables –A INPUT –j ACCEPT –p icmp –icmp-type 0
- iptables –A INPUT –j ACCEPT –p icmp –icmp-type 8
- iptables –A INPUT –j ACCEPT –p icmp –icmp-type 3
  - Enable ping functionality
- iptables –A INPUT –p tcp –dport 80 –j ACCEPT
  - Excepts all connections to port 80 from any host

Packet States

- Connection tracking
  - source, destination, and port
- Can be use to block NEW connection to internal network hosts.
  - iptables –A INPUT –m state –state NEW –i eth0 –j DROP
  - iptables –A INPUT –m state –state NEW ! –i eth0 –j ACCEPT
- Allow local system to maintain connections to Internet
  - **iptables –A INPUT –m state –state ESTABLISHED,RELATED –j ACCEPT**

Network Address Translation NAT

- To add rule to the NAT table you must specify it with the –t option
  - iptables –t nat
- There are two types of NAT operations
  - source NAT SNAT – SNAT target
    - Rules that alter source address
  - destination NAT DNAT – DNAT target
    - Rules that alter destination addresses

- Three chains used by the kernel for NAT table
  - PREROUTING is used by DNAT rules, these are packets arriving
  - POSTROUTING is used by SNAT rules, these are packets leaving
  - OUTPUT is used by DNAT rules for locally generated packets
- Turn on IP forwarding
- in /etc/sysctl.conf
  - net.ipv4.ip_forward = 1
- from the command line
  - echo 1 > /proc/sys/net/ipv4/ip_forward

- Masquerading
  - the process of using the IP address of the internet facing network device for all client traffic. All the local host masquerade as if their IP address is that of the internet connect device.
  - iptables –t nat –A POSTROUTING –o eth0 –j MASQUERADE

- Masquerading (specific hosts)
  - There is a one to one translation between a fully qualified IP Address and a private IP address behind the firewall
  - iptables –t nat –A PREROUTING –d 10.0.0.2 --to-destination 192.168.0.5 –j DNAT
  - iptables –t nat –A POSTROUTING –s 192.168.0.5 --to-source 10.0.0.2 –j SNAT