SRT210

Firewall, Netfilter and iptables

Raymond Chan
May 2014
Firewall

What is a firewall?
Firewall

A product that inspects
Network traffic
at the network and transport layers
of the OSI Reference Model
and
Makes pass or filter decisions.
[ by most peoples]
Firewall Implementation under Linux

Kernel: Netfilter
User Land: iptables
Netfilter

What is it?

A framework within the Linux kernel that can be used to hook functions into the networking stack at various stages.
Netfilter

What does it provide?

Mainly packet

Filtering

and

mangling
Netfilter

What does it provide?

But also capable of doing connection tracking and logging
iptables

What is it?

userland tools to tell the appropriate netfilter modules what action to take for matching packets
Netfilter modules

Available features:

- Packet filtering
  - Stateful packet filtering
  - Filter packets based on packet headers
  - Filter outgoing packets based on user ID
- Address Translation: S-NAT, D-NAT
- IP accounting and mangling
Iptables: tables, chains and rules

Rules – packet specification and action

• For matching specific packet
• Action to be taken if matched

Chains

• Related rules are placed in the same chain
• Order of rule in chain a vital

Tables

• Chain of rules for the same function are placed in specific tables
Iptables: tables

Tables

- filter table – *all the firewall rules live here*
- nat table – all the rules for Network Address Translation
- mangle table – all the rules for altering packets
- raw table – for advanced connection tracking
Iptables: chains

Chains

- **Built-in chain**
  - INPUT (filter, mangle)
  - OUTPUT (filter, nat)
  - FORWARD (filter, mangle)
  - PREROUTING (nat, mangle, raw)
  - POSTROUTING (nat, mangle)

- **User defined chain**
Iptables: tables and chains

Table:
- filter
- nat
- mangle

Chain:
- input
- output
- forward
- prerouting
- postrouting
- user-defined
Iptables: rules

Default Rule – Chain Policy
- Built-in chain
- Rule of last resort

Regular Rules
- Matches expression
- Target – action to be taken
Iptables: rules

Matches:

- **--source** (-s)
- **--destination** (-d)
- **--protocol** (-p)
- **--in-interface** (-i)
- **--out-interface** (-o)
- **--state**
- **--string**
Iptables: rules

Targets:

- ACCEPT
- DROP
- LOG
- REJECT
- RETURN
- QUEUE
Iptables: rules

Targets

- ACCEPT – let the packet through
- DROP – drops the packet
- REJECT – drops the packet and send back an error packet in response to the matched packet
- QUEUE – passes the packet to userspace (application program)
- RETURN – stops traversing the current chain and resume at the next rule in the previous (calling) chain
Iptables: chain policy

- Either **ACCEPT** or **DROP**
- Policy of a chain applied when
  - A packet reaches the end of a built-in chain, or
  - A packet matches a rule in a built-in chain with the target RETURN
Iptables: tables and chains

Filter Table's built-in chains:

- INPUT chain
- OUTPUT chain
- FORWARD chain
Iptables: tables and chains

NAT Table's built-in chains:

- PREROUTING chain
  - Translate destination addresses
- OUTPUT chain
- POSTROUTING chain
  - Translate source addresses
Iptables: tables and chains

Mangle Table's built-in chains:

- PREROUTING chain
- OUTPUT chain
- INPUT chain
- FORWARD chain
- POSTROUTING chain
Firewall: Basic Operation

- Creating firewall rules (*in filter table*)
- Listing existing firewall rules
- Flushing out existing firewall rules
- Set/Change chain policy
- Saving existing firewall rules to file
- Restore firewall rules from file
Firewall: Basic Operation

Chain Policy

- To list current chain policy (default to filter)
  - `iptables [-t table] -L`
- To set chain policy for chains in the filter table (DROP or ACCEPT)
  - `iptables -P INPUT DROP`
  - `iptables -P OUTPUT ACCEPT`
  - `iptables -P FORWARD ACCEPT`
Firewall: Basic Operation

Chain Policy

- Each chain must be set individually
- OUTPUT chain in the filter table and the OUTPUT chain in the nat table are separate chains
- Use the "-t table" to specify which table the chain is in, default to filter table if not specified
- User defined chain does not need a chain policy
Firewall: Basic Operation

A Perfectly secure Firewall

- `iptables -t filter -F`
- `iptables -t filter -P INPUT DROP`

Note: `-F` only flushes all the rules in all the chains in a given table, the current chain policy remain unchanged. After executing the above two commands, netfilter will drops all incoming packets.
Firewall: Basic Operation

A Practically secure Firewall

- `iptables -t filter -F`
- `iptables -t filter -P INPUT DROP`

Allow packets from any web server to go through the firewall:

- `iptables -I INPUT -p tcp --sport 80 -j ACCEPT`

Allow incoming SSH connection requests from any machine:

- `iptables -I INPUT -p tcp --port 22 -j ACCEPT`

Allow any incoming packet to the loopback interface (lo):

- `iptabled -I INPUT -i lo -j ACCEPT`
Firewall: Basic Operation

More firewall rules

Allow packets which responding to previous outgoing packets

- `iptables -I INPUT -m state --state RELATED,ESTABLISH -j ACCEPT`

Allow new incoming SSH connection requests from any machine

- `iptables -I INPUT -p tcp --port 22 -m state --state NEW -j ACCEPT`

Deny new incoming SSH connection requests from machine with IP address 192.168.99.65

- `iptabled -I INPUT -p tcp --port 22 -s 192.168.99.65 -m state --state NEW -j DROP`
Firewall: Basic Operation

More firewall rules

Reject any incoming ping packets

- `iptables -I INPUT -p icmp -j REJECT`

Drop any incoming ping packets

- `iptables -I INPUT -p icmp -j DROP`

Allow incoming ping packets only from 192.168.99.40

- `iptabled -I INPUT -p icmp -s 192.168.99.40 -j ACCEPT`
Firewall: Basic Operation

iptables rule syntax:

- `iptables [-t tables] [options] chain [match] [target]`
- `iptables [-t table] [-ADC] chain rule-specification [options]`
- `iptables [-t table] -I chain [rulenum] rule-specification [options]`
- `iptables [-t table] -R chain rulenum rule-specification`
- `iptables [-t table] -D chain rulenum [options]`
- `iptables [-t table] -{LFZ} [chain] [options]`
- `iptables [-t table] -N chain`
- `iptables [-t table] -X [chain]`
- `iptables [-t table] -P chain target [options]`
- `iptables [-t table] -E old-chain-name new-chain-name`
MAC match rule:

- **iptables [-t tables] [options] chain [match] [target]**
  
  Example (append rules to filter table):

- **iptables -A INPUT -m mac --mac-source 00:05:0A:1B:2D:3E -j ACCEPT**

- **iptables -A INPUT -m mac --mac-source ! 00:05:0A:1B:2D:3E -j ACCEPT**

- MAC match valid only in the PREROUTING, FORWARD and INPUT chains
Firewall: Basic Operation

**Save Current rules:**

```bash
iptables-save [-c] [-t table]
```

Dump the contents of an IP table in a parseable format to STDOUT.

```bash
iptables-save > firewall.txt
```

Dump all the current IP tables to the file firewall.txt
Firewall: Basic Operation

restore rules from file:

`iptables-restore [-c] [-n]`

Restore IP tables from data specified on STDIN

- `-c` restore the values of all packet and byte counters
- `-n` don't flush the previous contents of the table

`iptables-restore < firewall.txt`

Restore IP tables from the previously save file `firewall.txt`
Firewall
Netfilter & iptables

Questions & Answers

Raymond Chan
May 2014