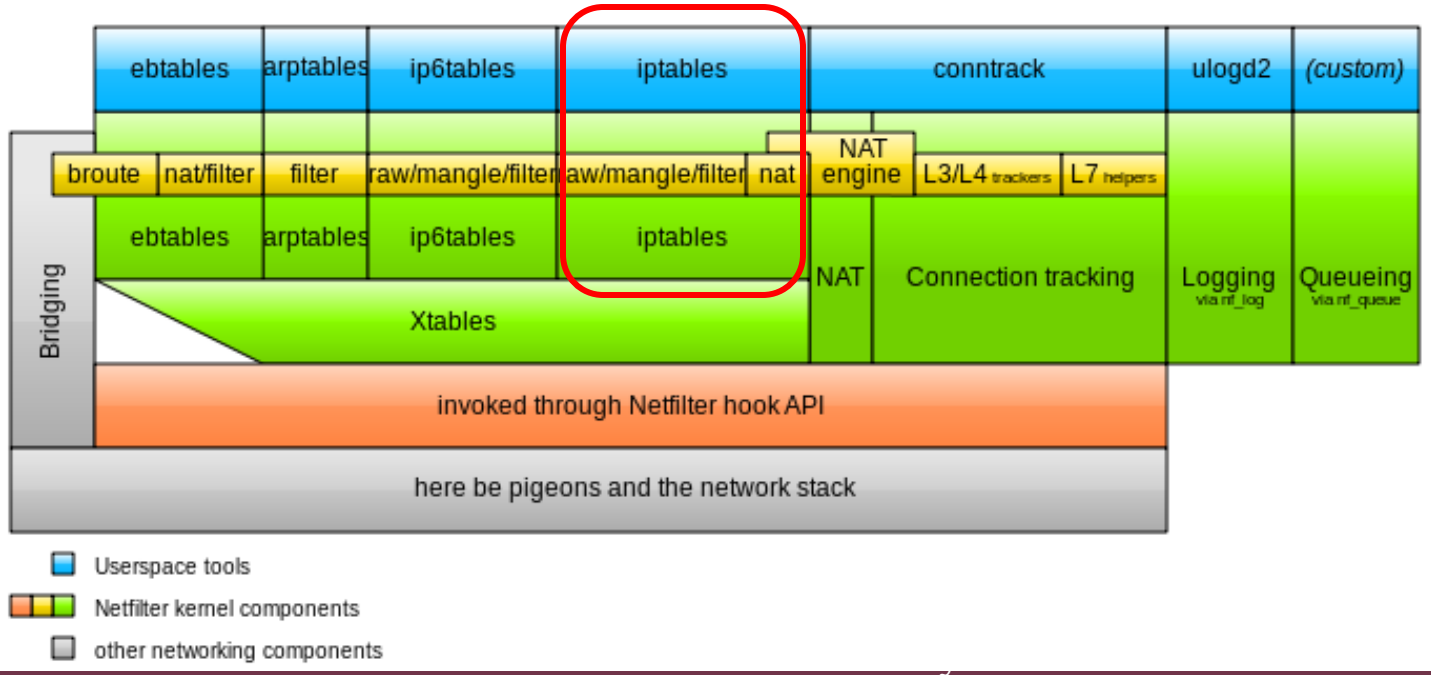
iptables

Chun-Jen (James) Chung

Arizona State University

What is iptables?

- It is a statful packet filtering firewall.
- It interfaces to the Linux *netfilter* kernel module to perform filtering of network packets based on a ruleset.

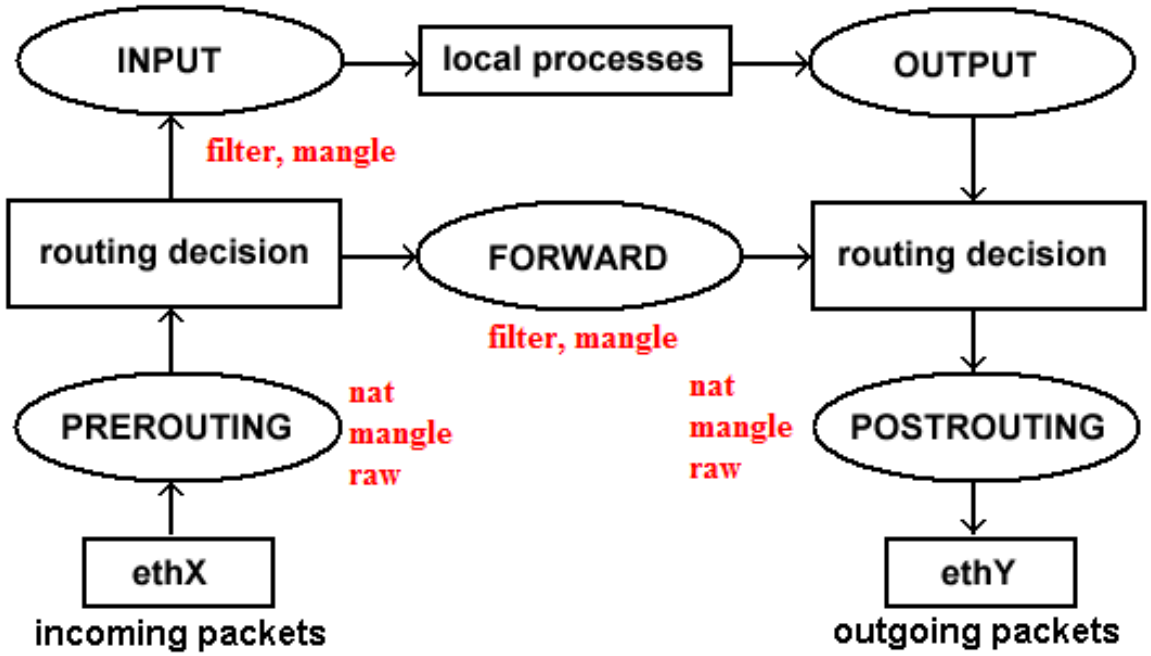


Four Major Tables

- iptables provide a table-based system for ***defining firewall rules*** that can filter or transform packets.
  - **raw**: rules used for keeping packets without be changed and that should not be handled by the connection tracking system.
  - **mangle**: rules used for modifying packet's TOS or TTL fields.
  - **nat**: rules used for translating the packet's src and dst field.
  - **filter**: rules used for filtering packets. (the default table)
- Each table contains one or more chain.

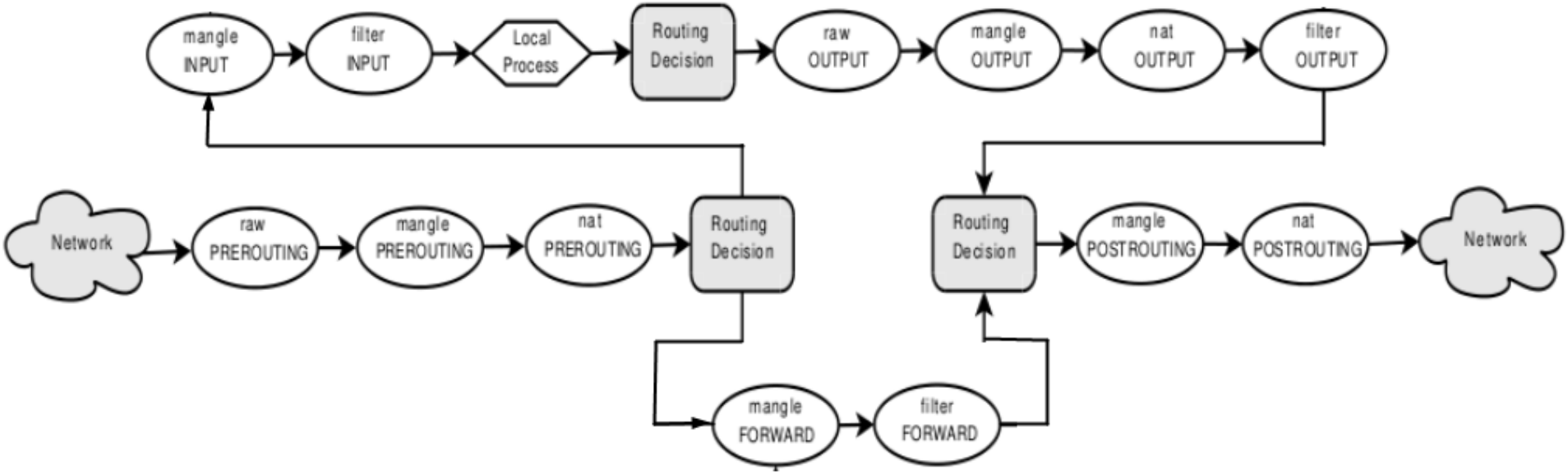
Five Predefined Chains

- Each table containing chains of rules for the treatment of packets.
  - PREROUTING:
  - INPUT:
  - FORWARD:
  - OUTPUT:
  - POSTROUTING:



- Predefined chains have a policy (DROP, ACCEPT)

Detailed Packet Path



raw: PREROUTING
INPUT, OUTPUT

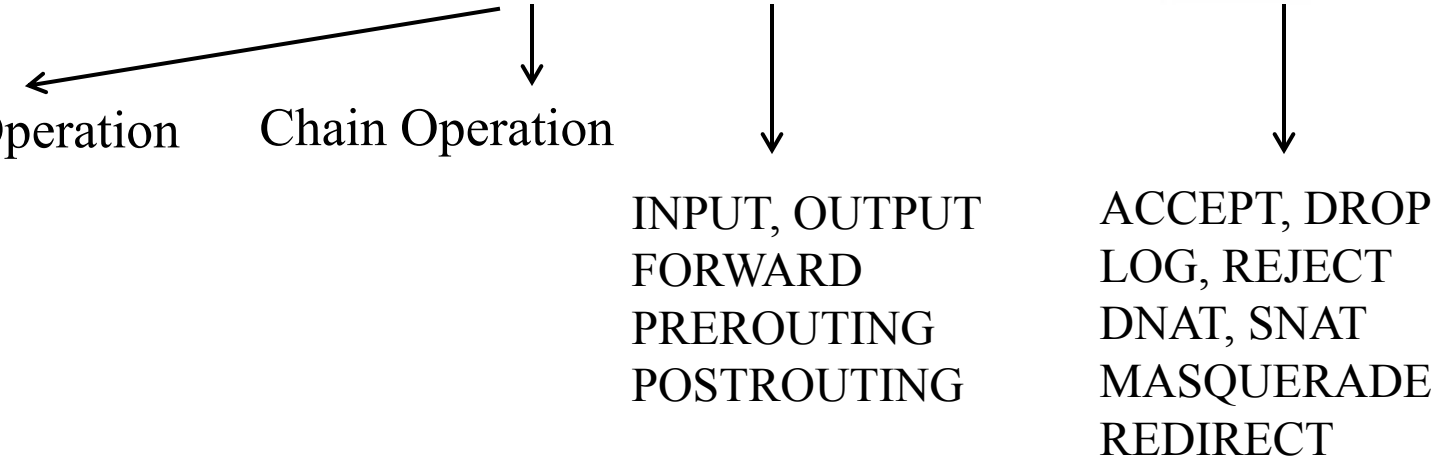
filter: INPUT, OUTPUT
FORWARD

nat: PREROUTING (DNAT/REDIRECT)
OUTPUT (DNAT/REDIRECT)
POSTROUTING (SNAT/MASQUERADE)

mangle: INPUT, OUTPUT, FORWARD
PREROUTING, POSTROUTING

Command Structure

```
iptables -t <table> operation chain specs -j target
```



- *Specs:* specify packet characteristics for matching
  - If the packet does not match the specification the packet is handed off to the next rule in the chain
  - If the packet meets the specification then the rule is passed to the `target`

Rule Operations

- `-I` Add a rule to the head of a chain
- **`-A`** Appends a rule to the tail of a chain
- `-D` Deletes a rule that matches the specifiers
- `-R` Replaces a rule in a chain

Syntax

```
iptables -t table -OP chain specifiers
iptables -t table -OP chain line# specifiers
iptables -OP chain specifiers
```

Chain Operations

- Listing a chain

```
iptables -t <table> -L chain
iptables -L chain
iptables -L
```

- Flushing a chain

Deletes all rules associated with a chain

```
iptables -t table -F chain
iptables -F chain
iptables -F
```

- Setting the default policy of a chain (filter)

```
iptables -P chain policy
```

policy → DROP, ACCEPT, REJECT, …

Chain Operations (for User Chain)

- Creating a user chain

```
iptables -t table -N chain
iptables -N chain
```

- Deleting a user chain

```
iptables -t table -X chain
iptables -X chain
iptables -X
```

- Renaming a user chain

```
iptables -t table -E old new
iptables -E old new
```

Specs: Packet characteristics

- Protocol
- Source IP
- Destination IP
- Input Interface
- Output interface
- Frag flag

- TCP Datagrams
  - Src port
  - Dst port
  - Flags
  - TCP options
- UDP Datagrams
  - Src port
  - Dst port
- ICMP Messages
  - Type and code

Protocol field

- Syntax: **`-p | --protocol [!] [<protocol>]`**
- Protocol name: `tcp,udp,icmp`
- Protocol number as list in `/etc/protocols`
  - all (0), icmp (1), tcp (6), udp (17), …
- Examples
  - `--protocol 0`
  - `-p tcp,udp`
  - `-p ! udp`
  - `--protocol icmp`

`icmp` Type and Code

RFC 792

- Syntax: **`-p icmp -icmp-type <type>`**
- Type (code):
  - `echo-request (0)`
  - `echo-reply (8)`
  - `destination-unreachable (3)`
  - `source-quench (4)`
  - `time-exceeded (10)`
  - `redirect (5)`

Source/Destination IP Address

- Syntax:

```
-s | --source | --src [!] <address>[/mask]
-d | --destination | --dst [!] <address>[/mask]
```

- Examples
  - `-s 1.2.3.4`
  - `-s 192.168.0.1/255.255.255.0`
    - IP address/network mask (Specifies a range of IP addresses)
  - `-s 192.168.0.0/24`
    - Specifies a range of addresses (192.168.0.0 – 192.168.0.255)
  - `-s ! 10.0.0.0/8`
    - Everything except `10.0.0.0-10.255.255.255`

Interface

- `-i` <Input interface>
  - Only in INPUT, FORWARD, PREROUTING chains
    - `-i eth0`
    - `-i ! eth0` except eth0
    - `-i eth+` all ethernet interfaces
    - `-i lo` loop back interface
- `-o` <Output interface>
  - Only in OUTPUT, FORWARD, POSTROUTING chains

Fragment

- Specifies second and additional fragmented packets. The negated version of this specifies unfragmented packets.
- Syntax:

  `[!] -f | --fragment`
- Examples
  - `-f` frag flag is set
  - `! -f` frag flag is not set

Port specs

- Syntax

  --source-port || --sport [!] <port>[:<port>]
  --destination-port || --dport [!] <port>[:<port>]

- examples
  - `-p tcp --sport 80`
  - `-p udp --dport 53`
  - `-p tcp,udp --sport 0:1023`
  - `-p tcp,udp --sport 1024`
  - `-p tcp,udp --dport 1024:`

TCP Flags

- `-p tcp –tcp-flags SYN,ACK,FIN SYN`
  - Tests SYN, ACK, FIN flags to see if the SYN bit is the only flag set
  - Possible flags
    - `ACK`
    - `FIN`
    - `RST`
    - `PSH`
    - `SYN`
    - `URG`

SYN

- Tests `tcp` packets for `SYN` to be set and `ACK` and `FIN` not set
  - `-p tcp --syn`

    Filters all packets requesting `tcp` connection (new connection)
  - `-p tcp ! --syn`

Connection State

- **`-m state --state <state-specifier>`**
- State-specifiers
  - `NEW` Associated with a new connection request
  - `ESTABLISHED` Associated with an established connection
  - `RELATED` Associated with a new secondary connection request related to an established connection (`ftp, icmp`)
  - `INVALID` Associated with a bad connection or is malformed
- examples:
  - iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
  - iptables -A OUTPUT --out-interface <interface> -p udp -s $IPADDR --source-port 53 -d $NAME_SERVER --destination-port 53 -m state --state NEW,RELATED -j ACCEPT

Rate Limits

- **`-m limit --limit <rate/unit>`**
- Rate/unit
  - `rate:` Packets per unit time
  - `Unit: Second, minute, hour, day`
- **`-m limit --time-burst number --limit <rate/unit>`**
  - `number` – max permitted burst before rate limit is applied
- Examples
  - iptables -A INPUT -i eth0 -p icmp --icmp-type echo-request -m limit --limit 1/second -j ACCEPT
  - iptables -A INPUT -i eth0 \ -p icmp --icmp-type echo-request -j DROP

Targets/Actions

- Target types
  - Firewall actions – filter table chains & user defined
    - » `ACCEPT, DROP, REJECT, LOG, RETURN`
  - NAT support
    - » `DNAT, MASQ, REDIRECT, SNAT`
  - Uncommon targets
    - » `MARK, MIRROR, QUEUE, TOS, TTL, ULOG`

Firewall Actions

- `iptables operation specification -j target`
  - If the packet does not match the specification the packet is handed off to the next rule in the chain
  - If the packet meets the specification then the rule is passed to the `target`

Firewall Actions

- `-j ACCEPT`
  - Lets the packet satisfying the specification **pass** to the next chain in the packet path
- `-j DROP`
  - The packet satisfying the specification is **dropped with no error packet sent to the sender**
  - *Stealth mode* – used for packet blocking on sensitive hosts

Firewall Actions

- `-j REJECT`
  - The packet satisfying the specification is dropped **with an error packet sent to the sender**
  - `-j REJECT` default error is **port unreachable**
  - `-j REJECT --reject-with` *`flag`*
    - *`icmp-net-unreachable`*
    - *`icmp-host-unreachable`*
    - *`icmp-port-unreachable`*
    - *`Icmp-proto-unreachable`*
    - *`icmp-net-prohibited`*
    - *`icmp-host-prohibited`*
    - *`tcp-reset`*
    - *Sends a tcp packet with the `RST` bit set*

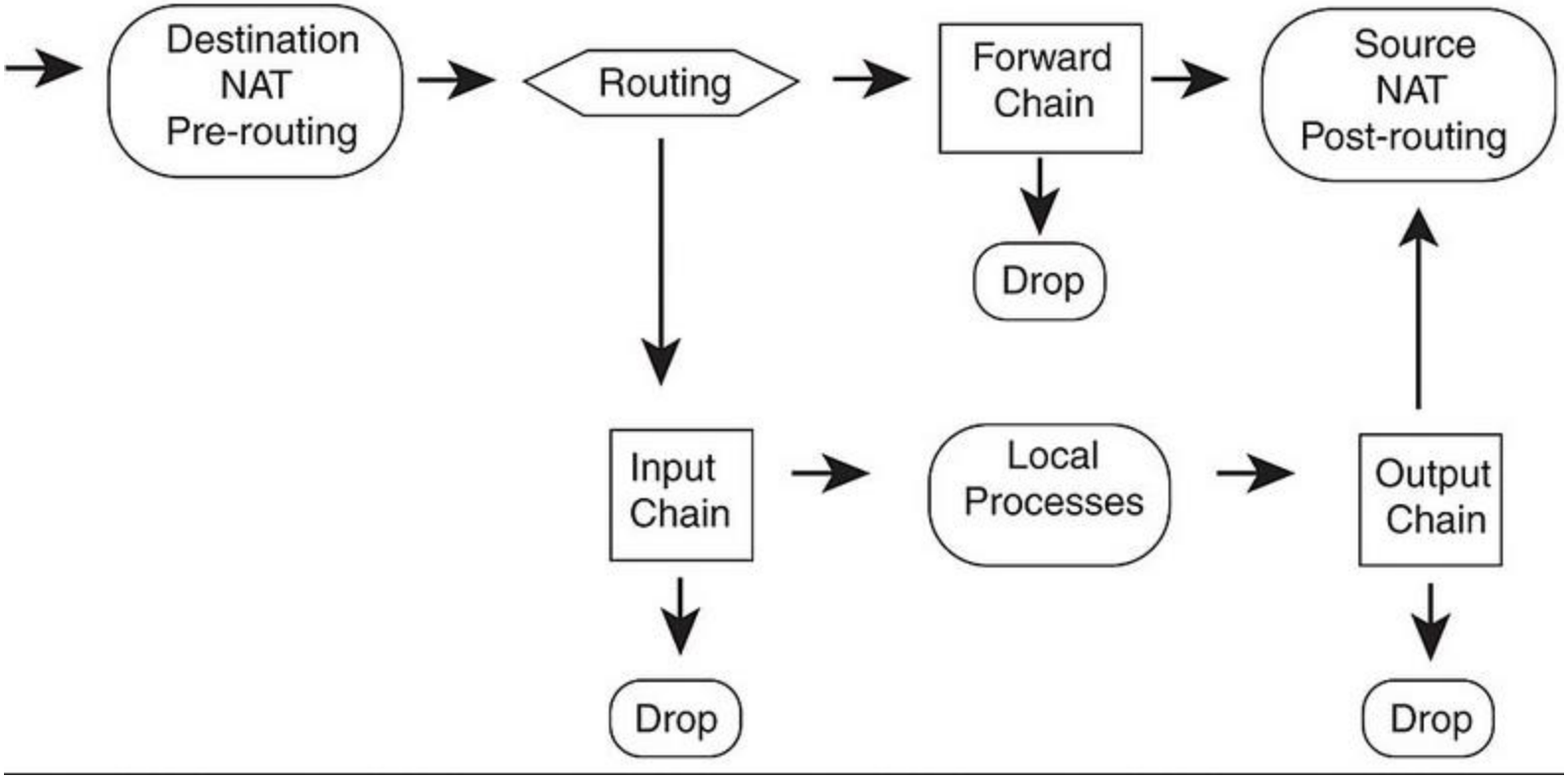
Firewall Actions

- `-j LOG`
  - Causes the packet satisfying the specification to be logged using the **Syslog** facility
  - `--log-prefix "IPT description of entry"`
    - IPT identifies the source of the log entry, i.e. Iptables
    - Description within quotes is limited to 29 characters
  - `--log-ip-options`
  - `--log-level`
  - `--log-tcp-options`
  - `--log-tcp-sequence`
  - To log a dropped packet a log rule must precede the dropping rule

Firewall Actions

- `-j user-chain-name`
  - Lets the packet satisfying the specification pass to the named user chain
- `-j RETURN`
  - Used in the user chain to return to the calling chain

NAT


Destination
NAT
Pre-routing
Routing
Forward
Chain
Source
NAT
Post-routing
Drop
Input
Chain
Local
Processes
Output
Chain
Drop
Drop

NAT Actions

- DNAT: rewrite the **destination IP** address of the packet
  - Iptables -t nat -A PREROUTING -p TCP -i eth1 -d $HTTP_IP --dport 80 -j DNAT --to-destination $DMZ_HTTP_IP
- SNAT: rewrite the **source IP** address of the packet
  - iptables -t nat -A POSTROUTING -o eth0 -j SNAT --to-source $INET_IP
- MASQ: rewrite the **source IP** address of the packet to the IP address of firewall's outgoing interface
  - iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE

Filter Actions

- INPUT
  - iptables -A INPUT -s 0/0 -i eth0 -d 192.168.1.1 -p TCP -j ACCEPT
  - iptables -A INPUT -p icmp --icmp-type echo-reply -j ACCEPT
- FORWARD
  - iptables -A FORWARD -s 0/0 -i eth0 -d 192.168.1.58 -o eth1 -p TCP -sport 1024:65535 -dport 80 -j ACCEPT
- OUTPUT
  - iptables -A OUTPUT -p icmp --icmp-type echo-request -j ACCEPT