

# Iptables with shorewall!

## Πίνακας περιεχομένων

1. Install swarmlab-sec (Home PC) .....	1
2. shorewall .....	1
2.1. Installation .....	2
3. Basic Two-Interface Firewall .....	2
4. Shorewall Concepts .....	3
4.1. zones — Shorewall zone declaration file .....	3
4.2. interfaces — Shorewall interfaces file .....	4
4.3. policy — Shorewall policy file .....	4
4.4. rules — Shorewall rules file .....	4
4.5. Compile then Execute .....	4
5. Three-Interface Firewall .....	5
5.1. zones .....	6
5.2. interfaces .....	6
5.3. policy .....	7
5.4. rules .....	7
5.5. masq - Shorewall Masquerade/SNAT definition file .....	7
5.6. snat — Shorewall SNAT/Masquerade definition file .....	8
5.7. Compile and Execute .....	8

## 1. Install swarmlab-sec (Home PC)

HowTo: See <http://docs.swarmlab.io/lab/sec/sec.adoc.html>



*NOTE*

Assuming you're already logged in

## 2. shorewall

**Shorewall** is an open source firewall tool for Linux that builds upon the Netfilter (iptables/ipchains) system built into the Linux kernel, making it easier to manage more complex configuration schemes by providing a higher level of abstraction for describing rules using text files.

More: [wikipedia](#)

*NOTE*

Our docker instances have only one nic

to add more nic's:

*create netowrk frist*

```
docker network create --driver=bridge --subnet=192.168.0.0/16 net1
docker network create --driver=bridge --subnet=192.168.0.0/16 net2
docker network create --driver=bridge --subnet=192.168.0.0/16 net3
```



then connect network to container

*connect network created to container*

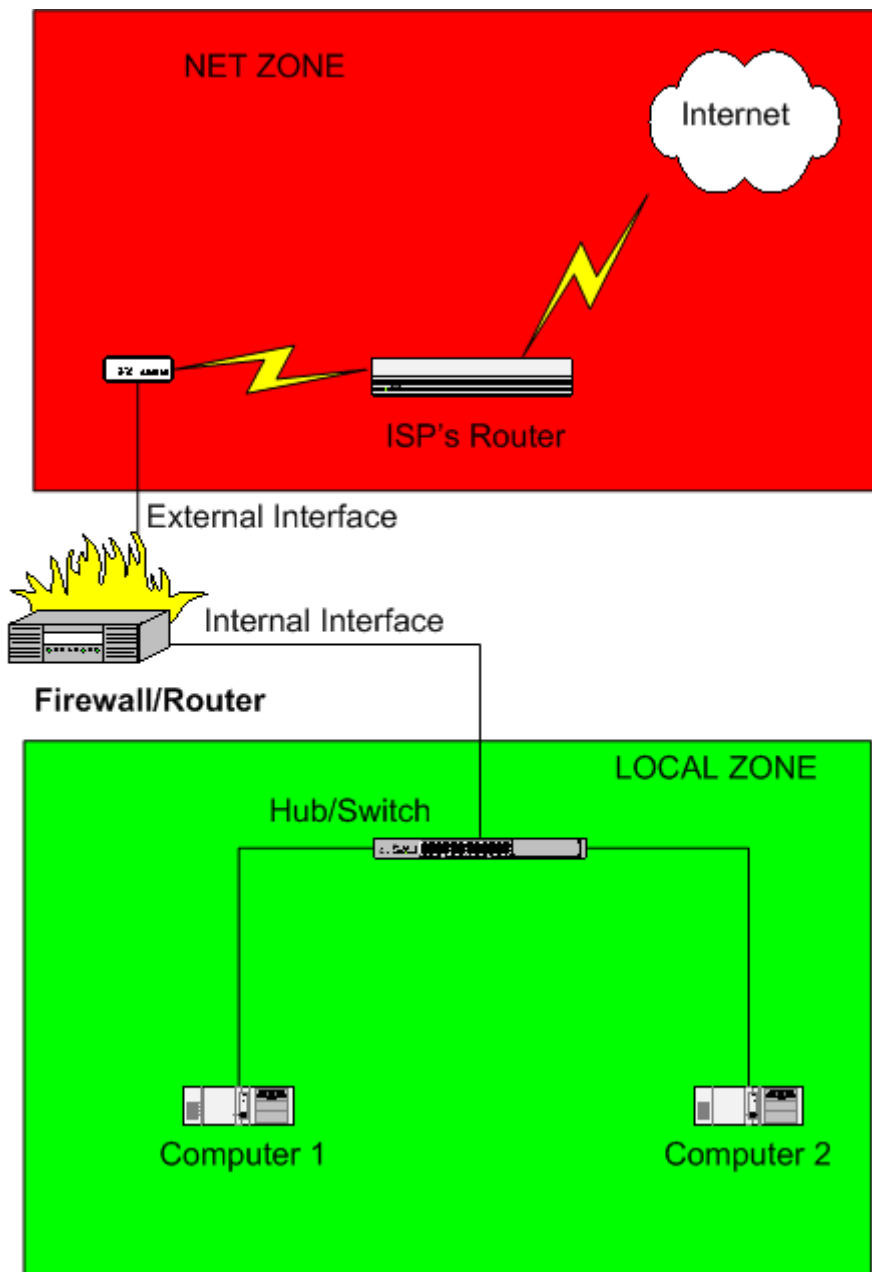
```
docker network connect net1 master
docker network connect net1 worker1
docker network connect net2 master
docker network connect net2 worker2
```

now let's look at the following image

## 2.1. Installation

Shorewall is already installed on swarmlab-sec.

## 3. Basic Two-Interface Firewall



*connect to master first*

Assuming you're already logged in master!

master is now our Firewall/Router

swarmlab-sec login

## 4. Shorewall Concepts

The configuration files for Shorewall are contained in the directory `/etc/shorewall`

### 4.1. zones — Shorewall zone declaration file

The `/etc/shorewall/zones` file declares your network zones. You specify the hosts in each zone through entries in `/etc/shorewall/interfaces`

*/etc/shorewall/zones*

#ZONE	TYPE	OPTIONS	IN_OPTIONS	OUT_OPTIONS
fw	firewall			
net	ipv4			
loc	ipv4			

## 4.2. interfaces — Shorewall interfaces file

The interfaces file serves to define the firewall's network interfaces to Shorewall.

*/etc/shorewall/interfaces*

#ZONE	INTERFACE	BROADCAST	OPTIONS
net	eth0		dhcp,routefilter
loc	eth1	detect	

## 4.3. policy — Shorewall policy file

This file defines the high-level policy for connections between zone

*/etc/shorewall/policy*

#SOURCE	DEST	POLICY	LOGLEVEL	LIMIT
loc	net	ACCEPT		
net	all	DROP	info	
all	all	REJECT	info	

## 4.4. rules — Shorewall rules file

Entries in this file govern connection establishment by defining exceptions to the policies

*/etc/shorewall/rules*

#ACTION	SOURCE	DEST	PROTO	DPORT
ACCEPT	\$FW	net	udp	53
ACCEPT	net	\$FW	udp	53
ACCEPT	\$FW	net	tcp	80
ACCEPT	net	\$FW	tcp	80

## 4.5. Compile then Execute

Shorewall uses a "compile" then "execute" approach. The Shorewall configuration compiler reads the configuration files and generates a shell script. Errors in the compilation step cause the script to be discarded and the command to be aborted. If the compilation step doesn't find any errors then the shell script is executed.

```
/sbin/shorewall start  
/sbin/shorewall stop  
/sbin/shorewall clear
```

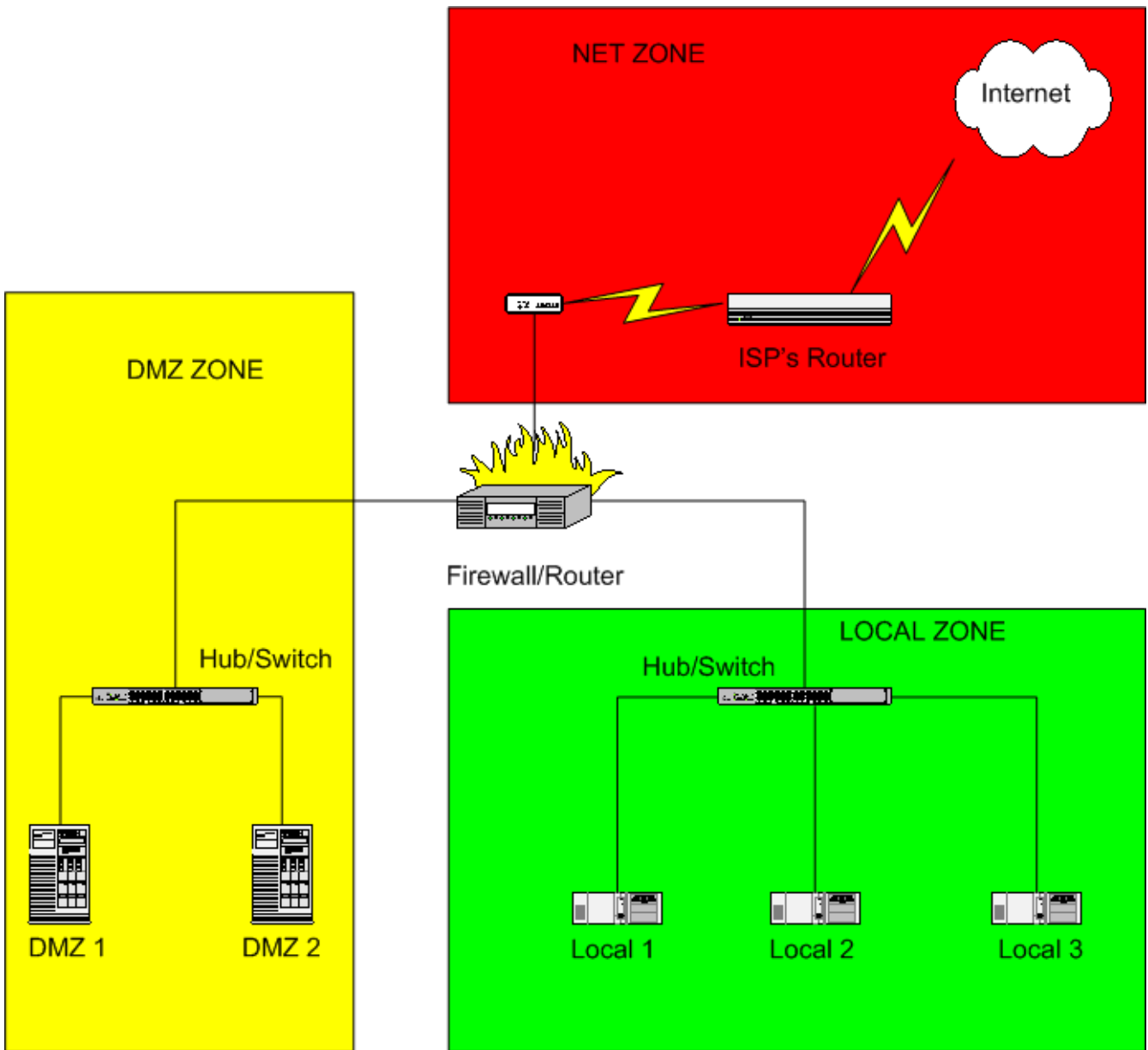
*NOTE*

The 'compiled' scripts are placed by default in the directory `/var/lib/shorewall` and are named to correspond to the command being executed. For example, the command `/sbin/shorewall start` will generate a script named `/var/lib/shorewall/.start` and, if the compilation is error free, that script will then be executed. If the script executes successfully, it then copies itself to `/var/lib/shorewall/firewall`. When an `/sbin/shorewall stop` or `/sbin/shorewall clear` command is subsequently executed, `/var/lib/shorewall/firewall` is run to perform the requested operation.



The `AUTOMAKE` option in `/etc/shorewall/shorewall.conf` may be set to automatically generate a new script when one of the configuration files is changed. When no file has changed since the last compilation, the `/sbin/shorewall start`, `/sbin/shorewall reload` and `/sbin/shorewall restart` commands will simply execute the current `/var/lib/shorewall/firewall` script.

## 5. Three-Interface Firewall



## 5.1. zones

*/etc/shorewall/zones*

```
#ZONE  TYPE  OPTIONS                IN_OPTIONS  OUT_OPTIONS
fw      firewall
net     ipv4
loc     ipv4
dmz     ipv4                #new line
```

## 5.2. interfaces

*/etc/shorewall/interfaces*

```
#ZONE    INTERFACE    BROADCAST    OPTIONS
net      eth0         dhcp,routefilter
loc      eth1         detect
dmz      eth2         detect        #new line
```

## 5.3. policy

*/etc/shorewall/policy*

```
#SOURCE  DEST        POLICY    LOGLEVEL    LIMIT
loc      net        ACCEPT
dmz      net        DROP      #new line
net      all        DROP      info
all      all        REJECT    info
```

## 5.4. rules

*/etc/shorewall/rules*

```
#ACTION  SOURCE  DEST        PROTO  DPORT
ACCEPT  $FW    net        udp    53
ACCEPT  net    $FW        udp    53
ACCEPT  $FW    net        tcp    80
ACCEPT  net    $FW        tcp    80
#New Lines
ACCEPT  $FW    dmz        udp    53
ACCEPT  dmz    $FW        udp    53
ACCEPT  $FW    dmz        tcp    80
ACCEPT  dmz    $FW        tcp    80

ACCEPT  loc    dmz        tcp    80 # Add your rules for the zones
you have defined.
ACCEPT  dmz    loc        tcp    80 #
ACCEPT  loc    net        tcp    80 # This here is an example
ACCEPT  net    loc        tcp    80 # for communication
ACCEPT  dmz    net        tcp    80 # over port 80
ACCEPT  net    dmz        tcp    80 # aka the web
```

## 5.5. masq - Shorewall Masquerade/SNAT definition file

*/etc/shorewall/masq* - directs the firewall where to use many-to-one (dynamic) Network Address Translation (a.k.a. Masquerading) and Source Network Address Translation (SNAT).

*/etc/shorewall/masq*

```
#INTERFACE  SOURCE          ADDRESS          PROTO  DPORT
eth0        eth1
eth0        eth2
```

## 5.6. snat — Shorewall SNAT/Masquerade definition file

This file is used to define dynamic NAT (Masquerading) and to define Source NAT (SNAT). It superseded shorewall-masq(5) in Shorewall 5.0.14.

*/etc/shorewall/masq*

```
#ACTION  SOURCE          DEST
MASQUERADE 192.168.0.0/24  eth0
MASQUERADE 192.168.1.0/24  eth0
```

- You have a simple masquerading setup where eth0 connects to internet and eth1 connects to your local network with subnet 192.168.0.0/24.
- You add a router to your local network to connect subnet 192.168.1.0/24 which you also want to masquerade. You then add a second entry for eth0 to this file



Beginning with that release, the Shorewall compiler will automatically convert existing masq files to the equivalent snat file, and rename the masq file to masq.bak.

## 5.7. Compile and Execute

*/sbin/shorewall*

```
/sbin/shorewall start
/sbin/shorewall stop
/sbin/shorewall clear
```





*Reminder*

Caminante, no hay camino,  
se hace camino al andar.

Wanderer, there is no path,  
the path is made by walking.

**Antonio Machado** Campos de Castilla