Example of a basic lock down of the LAN and DMZ out going rules

Outbound LAN

- Make sure the “Default LAN > any” rule is either disabled or removed.
- Allow DNS access - if pfSense is the DNS server, use LAN address, if using outside DNS create rule to allow TCP/UDP 53 to anywhere
    - Allow TCP/UDP 53 (DNS) from LAN subnet to LAN Address, -or-
    - Allow TCP/UDP 53 (DNS) from LAN subnet to Upstream DNS Servers, -or-
    - Allow TCP/UDP 53 (DNS) from LAN subnet to anywhere
- Allow all users to browse web pages anywhere.
    - Allow TCP 80 (HTTP) from LAN subnet to anywhere
- Allow users to browse secure web pages anywhere.
    - Allow TCP 443 (HTTPS) from LAN subnet to anywhere
- Allow users to access FTP sites anywhere.
    - Allow TCP 21 (FTP) from LAN subnet to anywhere
- Allow users to access SMTP on a mail server somewhere.
    - Allow TCP 25 (SMTP) from LAN subnet to anywhere
- Allow users to access POP3 on a mail server somewhere.
    - Allow TCP 110 (POP3) from LAN subnet to anywhere
- Allow users to access IMAP on a mail server somewhere.
    - Allow TCP 143 (IMAP) from LAN subnet to anywhere
- To allow remote connections to an outside windows server, configure a rule for Remote administration.
    - Allow TCP/UDP 3389 (Terminal server) from LAN subnet to **IP address of remote server**
- To allow LAN to access windows shares on the DMZ, allow NETBIOS/Microsoft-DS from the LAN to the DMZ
    - Allow TCP/UDP 137 from LAN subnet (NETBIOS) to **DMZ subnet**
    - Allow TCP/UDP 138 from LAN subnet (NETBIOS) to **DMZ subnet**
    - Allow TCP/UDP 139 from LAN subnet (NETBIOS) to **DMZ subnet**
    - Allow TCP 445 from LAN subnet (NETBIOS) to **DMZ subnet**

Outbound DMZ

- By default, there are no rules on OPT interfaces.
- To allow servers to use Windows update or browse the WAN
    - Allow TCP 80 from DMZ subnet (HTTP) to anywhere
    - Allow TCP 443 from DMZ subnet (HTTP) to anywhere
- If an external DNS server is used, allow the computers to leave the network to connect to a DNS server.

        - Allow TCP/UDP 53 from DMZ subnet (DNS) to **IP address of the upstream DNS server (s)**
- To allow servers to use a remote time server open UDP port 123
    - Allow UDP 123 from DMZ subnet (NTP) to **IP address of remote time server** -or-
    - Allow UDP 123 from DMZ subnet (NTP) to any

Example setup isolating LAN and DMZ but each with unrestricted Internet access

The strict approach above may not be necessary if outbound access should be more lenient, but still controlled between local interfaces. The following setup can be used instead.

Prerequisites/Assumptions

This assumes all local networks are privately numbered, and that interfaces have already been configured.

Create an alias (**Firewall > Aliases**) called *RFC1918* containing *192.168.0.0/16*, *172.16.0.0/12*, and *10.0.0.0/8*

LAN Configuration

- Allow TCP/UDP from LAN subnet to LAN Address port 53 for DNS from the firewall
- Allow TCP from LAN subnet to LAN address port 443 for accessing the GUI
- Allow ICMP from LAN subnet to LAN address to ping the firewall from the LAN
- Allow any traffic required from LAN to DMZ (if any)
- Reject Any from LAN subnet to RFC1918 -- Do not allow LAN to reach DMZ or other private networks
- Allow Any from LAN subnet to any -- Internet access rule

DMZ Configuration

- Allow TCP/UDP from DMZ subnet to DMZ Address port 53 for DNS from the firewall
- Allow TCP from DMZ subnet to DMZ address port 443 for accessing the GUI (optional)
- Allow ICMP from DMZ subnet to DMZ address to ping the firewall from the DMZ
- Allow any traffic required from DMZ to LAN (if any)
- Reject Any from DMZ subnet to RFC1918 -- Do not allow DMZ to reach LAN or other private networks
- Allow Any from DMZ subnet to any -- Internet access rule

VyOS

Last modified: 2014-12-12 10:21:19

Querying system information

Action	Command
IP configuration	`show interfaces`
Routing	`show ip route`
Show configuration	`show`
Show log	`monitor log`, `show log tail`
Show IP traffic	`monitor interfaces`

CLI Modes

- operational mode (prompt `$`): view system status
- configuration mode (prompt `#`): modify system configuration

In configuration mode, you can execute "operational" commands by preceding them with `run`.

Basic configuration

Workflow:

```
vyos@vyos $ configure
vyos@vyos # [configuration commands]
vyos@vyos # commit
vyos@vyos # save
```

```
vyos@vyos # exit
vyos@vyos $
```

Action	Command
Set host name	set system host-name HOSTNAME
Set default gateway	set system gateway-address 192.168.0.1
Set DNS server	set system name-server 8.8.8.8
Turn on SSH access	set service ssh listen-address 0.0.0.0
Keyboard layout[^1]	sudo dpkg-reconfigure keyboard-configuration
Set time zone	set system time-zone [TAB]

[^1]: Use in non-config mode

Configuring network interfaces

Action	Command
Run "normal" commands in config mode	run COMMAND
Set IP address on interface	set interfaces ethernet eth0 address 192.168.0.1/24
Run DHCP client on interface	set interfaces ethernet eth0 address dhcp
Set interface description	set interfaces ethernet eth0 description WAN

Static routing

Action	Command
Add route	set protocols static route 192.168.0.0/24 next-hop 10.0.0.1 distance 1
Set default route	set protocols static route 0.0.0.0/0 next-hop 10.0.2.2 distance 1
Drop traffic	set protocols static route 172.16.0.0/12 blackhole distance '254'

RIP

Example with two directly connected networks:

```
# set protocols rip network 192.168.0.0/24
# set protocols rip network 192.168.1.0/24
# set protocols rip redistribute connected
```

Network Address Translation (NAT)

The following example adds a NAT rule with id 100 for a router with its WAN port on `eth0`. All IP addresses on the internal network 192.168.0.0/24 are translated into the router's IP address on `eth0`.

```
# set nat source rule 100 outbound-interface 'eth0'
# set nat source rule 100 source address '192.168.0.0/24'
# set nat source rule 100 translation address 'masquerade'
```

If you have multiple networks on the "inside", add a separate rule with a different id (e.g. 200).

DNS forwarding

Use DNS forwarding if you want your router to function as a DNS server for the local network. There are several options, the easiest being 'forward all traffic to the system DNS server(s)' (defined with `set system name-server`):

```
# set service dns forwarding system
```

Manually setting a DNS server for forwarding:

```
# set service dns forwarding name-server 8.8.8.8
# set service dns forwarding name-server 8.8.4.4
```

Setting a forwarding DNS server for a specific domain:

```
# set service dns forwarding domain example.com server 192.0.2.1
```

Example: router with two interfaces `eth0` (WAN link) and `eth1` (LAN). A DNS server for the local domain (example.com) is at 192.0.2.1, other DNS requests are forwarded to Google's DNS servers.

```
# set service dns forwarding domain example.com server 192.0.2.1
# set service dns forwarding name-server 8.8.8.8
# set service dns forwarding name-server 8.8.4.4
# set service dns forwarding listen-on 'eth1'
```

Script template

Use the following as a template for a configuration script:

```
#!/bin/vbash
source /opt/vyatta/etc/functions/script-template

configure

# Fix for error "INIT: Id "TO" respawning too fast: disabled for 5 minutes"
delete system console device ttyS0
```

```
# Commands here
```

commit
save