search...

Home | Fulup | Dominig | Site Map | Search

MAIN MENU

Home

FAQs & TIPs

Tools & Utilities

Links

VM-firewall - Virtualization firewall

Monday, 17 November 2008 20:50 Fulup Ar Foll

While they are many available firewall for Linux, none of them are really designed for virtualization. Obviously it is alway possible to hack an existing one to fit your need, but VM-firewall has been designed to support virtualization in a Internet hosted environment, it runs the same rules for Xen, OpenVZ and VirtualBox. The goal of VM-Firewall if to provide to virtualization administrator the same facility has the one provided by an intelligent router in a traditional architecture, it is particularly useful for people who have multiple IP addresses that they want to forward on different virtual machines.

Disclaimer

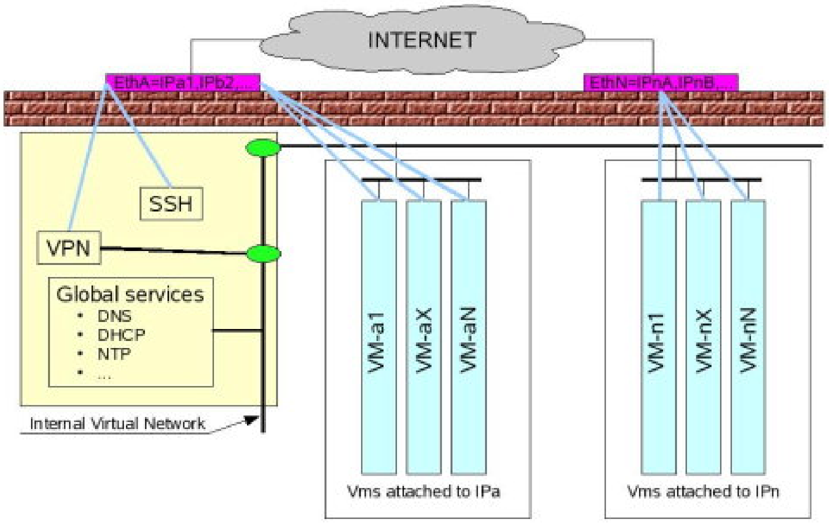
Anything I wrote here was done outside of my professional work context and none of my current/past employers/customers have participate or even be consulted for this work. Fridu is 100% part of my free time, and everything including hosting is funded on our pocket money and used to support non commercial friend organizations. While I think I have the technical background to design a smart architecture (cf:my profile). I nevertheless do not guaranty that it will work for you, or even that you will agree with me. I still hope it may help some of you and I would be more than happy to incorporate improvement if ever you have some.

Table of Contend

Architecture

When running virtualization in a hosting environment, the main issue is the lack of access to networking infrastructure. Usually you get access to a limited number of public IP, with no DHCP and/or ARPA facilities. This imposes your virtualization infrastructure to be completely invisible from your provider's router. For this reason VM-firewall does not expose virtual machines directly to physical interface, but leverages virtual network interfaces, iptables and routing capabilities to map data stream to a given VM.

> **For input initiated stream**: VM-firewall maps an external IP+port to a given destination VM+port. VM receives the original Internet source IP addresses and can very easily applies their own network access control. Note: one given VM may received internet stream from more than one external IP addr *(which does not mean it is a good idea)*
>
> **For output initiated stream**: VM-firewall groups VM into zones, Then based on zones internal IP addresses and attached virtual network interface, it reverses map VM's internal IP to chosen external IP address. Outgoing stream are handle through a simple NAP operation, the only difference with a simple NAT as your home aDSL modem/router is that depending on the selected zone, your parquet will appear as coming from a different public IP. Note that as external public IP address selection is based on zone local network internal IP+virtual-NIC, for a given VM, outgoing stream can only NAT one public IP addr.



VM-FIREWALL USES 3 CLASS OF OBJETS.

zones: virtual machines are grouped into zones. A configuration have two or more zones: a dummy one named "none" for the hypervisor, and one to many for virtual machines. Typically the number of zones is equivalent to the number of external IP addresses+1(for hypervisor). Each zone may contend one to many virtual machines. zone is a set of virtual machines that shared a unique:

virtual network interface
local virtual network mask
external IP address

application: inside a zone it defines a port-forwarding for external data stream to internal virtual machines. An application takes as input a port and as destination a virtual machine's virtual network's IP+port.

tuning: optional values that allow you to enforce some global policies at zone level. Tuning may be used to prevent a zone to send SMTP traffic, or to limit SSH access from a given network, etc.

Configuration

INTRODUCTION

Config file can either be a flat file or a directory containing one to many *.conf rules. VM-firewall uses a two passes architecture: first pass reads every config files and loads them into tables, second pass scans those tables and generates iptables rules. This two pass model provides a lot of flexibility to configuration syntax, it:

decorelates configuration syntax from iptables structure
validates configuration independently of file order
generates easy to read/understand iptables rules
allow a lot of debugging tools

```
5.2
5.2
5.
```

Internal and VPN traffic are unlimited in both direction.

Note: zones do not define security inside your VM. If needed each VM can define it own access rules (firewall, apache acl, ...). Security provided by your zone is more or less equivalent of LAN and port locking as provided by intelligent switch/routers (Alteon, Cisco, ...)

APPLICATION

As said previously default zone behavior is to lock any traffic from Internet. When hosting applications you want some Internet traffic to be routed from the external NIC to your VMs. For each given application, WM-firewall is doing tree things:

open given port/proto on external interface for the given IP (in case you have more than one IP on your NIC)

forward incoming traffic to the internal VM using DNAT functionality

NAT outgoing traffic to to hide your Internal VM IP addr make it appear as coming from a public IP address.

```
# Example of 5 applications rule, sitting in our tree previsouly created zones
# ------------------------------------------------------------------------------------------
 CreateApp  NAME=SMTP           ZONE=Messaging      EXT=tcp:25        INT=10.10.1.2:25
 CreateApp  NAME=WEBMAIL        ZONE=Messaging      EXT=tcp:80        INT=10.10.1.3:80
 CreateApp  NAME=PORTAL         ZONE=Web            EXT=tcp:80        INT=10.10.2.2:8080
 CreateApp  NAME=PBX            ZONE=Messaging      EXT=udp:5600      INT=10.10.3.3:5600
 CreateApp  NAME=PRIVATEIP      ZONE=PRIVAT_IP      EXT=tcp:any       INT=10.10.4.3:any
```

For each application we have to provide

NAME a Label that should be unique in each file,

ZONE this is zone as defined previously, it will define you external IP address and your internal netmask.

EXT the external port/proto your application is listening. Be careful not to overlap your listening port, this especially when two zones are using the same external IP address.

INT your internal IP address and port, as obviously nothing force you to present the same port internally and externally.

TUNING

Optional zone access control parameters, that most of you will probably ignore, **expect if you're realy sure you need it, start without.** Nevertheless they are cases where not only you want to control external traffic, but you also want to control internal traffic. For example let's say that one of your zone is used by student for admin security labs, while some others are used for real "in production" university applications. You may want to restrict SSH access even from the internal LAN/VPN to your "in production zone". Equally, you may want to forbid peer to peer or SMTP outgoing from some zones. Tuning is important as soon as you cannot trust the admin of a given virtual machine, this even if this guy has root passwd for his own VM or even the full set of VM in a given zone. You as the hypervisor admin want to make sure than what ever any VM admin is doing you keep control or the global platform.

```
# Zone tuning (internal and outging traffic control)
# ---------------------------------------------------------------
  # Prevent Zb to send mail
    TuneZone  NAME=ZA_SMTP_NO   ZONE=ZA      DIR=out   ACTION=drop     PORT=tcp:25

  # Limit Zb SSH access to VPN subnet 10.10.11.0/255.255.255.0
    TuneZone  NAME=ZB_SSH_LIMIT ZONE=ZC  DIR=in ACTION=accept  PORT=tcp:22 src=10.10.11.

  # Only allow SSH and openVPN access to Zc
    TuneZone  NAME=ZC_SSH_OK    ZONE=ZC      DIR=in     ACTION=accept    PORT=tcp:22
    TuneZone  NAME=ZC_VPN_OK    ZONE=ZC      DIR=in     ACTION=accept    PORT=udp:1194
    TuneZone  NAME=ZC_ANY_NO    ZONE=ZC      DIR=in      ACTION=drop      PORT=tcp:any
```

Tuning applies in between opening application port and closing iptables chain of a given zone, this to assert that default action for incoming packets is "drop". Tuning iptable are applyied directly on zone Internal interface (ex: Xen Bridge, OpenVZ vnet) and can thus lock any traffic before it get the chance of behing evaluated for routing. You can use tuning rules to limit incoming/outgoing traffic, rules will affect both external an internal LAN.

Command line syntax

The parser is only a very basic shell script, and while my goal was to make the config file as simple as possible I did not spend much time in tracking lexical/grammatical errors. For this reason before going in production you MUST check that both you and my parser have the same understanding of your rules. This is especially important if you run a remote machine as locking your external interface may prevent you from accessing to your machine. General syntax is "Fridu-firewall.script command option-1=xxx ... option-n=..."

The parser will first read all your rules and then depending on your command will generate the adequate iptables.

Fridu-firewall.script display ;# config as understood by parser *[should be used after each rules update]*
Fridu-firewall.script test ;# start firewall and stop it automatically after 180s
Fridu-firewall.script start ;# build iptables and activate firewall
Fridu-firewall.script stop ;# delete every zone table and leave all your ports open with routing active
options:

1. config=file|directory ;# overwrite default config
2. debug=1 ;# only design to debug the script
3. verbose=1 ;# step by step verbose mode
4. dummy=1 ;# generate rules but do not apply them
5. dump=file ;# dump the iptable to a file

ex: Fridu-firewall.script **start dump=/**tmp/my-iptables *[Start firewall and dump a copy of rules in /tmp/my-iptables]*
ex: Fridu-firewall ***display config=****/etc/sysconfig/Fridu-firewall/samples/Fridu-firewall-Guru.config*

ACTION SECTION

This is optional, and hopefully most of you wont need it 🙂 Action section is not handle directly by VM-firewall script, it is only a convenient place for addon functions, you would like to start with firewall boot from /etc/init.d time, as:

- bridge creation
- special rules to allow one zones to talk to the other
- allow VPN traffic
- etc.

Any shell command can be used, action is either "start" or "start", as stopping a firewall before shutting down a machine is quite useless, is a special action is needed most of you should only implement the "start" part of it.

Example-XEN:

Warning: In order to use VM-firewall with Xen you need to update Xen's network script, check full article [here]

- allow VPN(tun+) to/from traffic to zone(xen-br+)
- allow zones to zones traffic (-i xen-br+ -o xen-br)

```
if test "$ACTION" = "start" ; then
  DoIt iptables  -A after-forwarding -i xen-br+ -o xen-br+ -j ACCEPT   # allow VM to talk together
  DoIt iptables  -A after-forwarding -i tun+    -o xen-br+ -j ACCEPT   # allow VPN talk to zones
  DoIt iptables  -A after-input      -i tun+      -j ACCEPT            # allow VPN talk to dom0
  DoIt iptables  -A after-forwarding -i xen-br+ -o tun+    -j ACCEPT   # allow Zones talk to VPN
fi
```

Example-OpenVZ:

For further information check OpenVZ-Proxmox and VM-firewall post [here]

- allow to/from VPN traffic(tun+) to zones virtual interface (venet+)
- allow zones to zones traffic (-i venet+ -o venet+)
- hack to map IP-two ssl port to internal 563 openvpn TCP port (this because SSL on IP-one is used for web applications)

```
# User Before/After Zone Custom Tables (before-input|output|forwarding, after-input|...)
# ---------------------------------------------------------------------------------
if test "$ACTION" = "start" ; then
  # DoIt modprobe  -s ip_conntrack_ftp                              # load FTP session tacking

  # we're not a bank make our life simple
  DoIt iptables  -A after-forwarding -i venet+ -o venet+  -j ACCEPT   # allow VM to talk together
  DoIt iptables  -A after-input      -i tun+    -j ACCEPT             # allow VPN talk to dom0
  DoIt iptables  -A after-forwarding -i tun+    -o venet+  -j ACCEPT  # allow VPN talk to zones
  DoIt iptables  -A after-forwarding -i venet+ -o tun+     -j ACCEPT  # allow Zones talk to VPN

  # Make SSL on IP-two to be redirected on port 563
  DoIt iptables  -A PREROUTING -t nat -i eth0 --destination 87.98.139.141 --proto tcp --dport 443 -j DNA
fi
```

Example VirtualBox

For further information on VirtualBox and VM-firewall check [here]

- create a bridge for virtual box and map local virtual address onto it
- create a Tun network interface for each virtual machine and add them into the bridge
- at stop time delete the bridge *[just to prove it works :)]*

```
if test "$ACTION" = "start" ; then

  # which use own virtualbox bridge
  VBOX_USER=fulup

  # zoneONE  bridge and tap
  brctl addbr vbox-br1 2>/dev/null
  ifconfig vbox-br1 $IP_BR1 netmask 255.255.255.0
  for VBOX in vbox12 vbox13
  do
    VBoxTunctl -u $VBOX_USER -t $VBOX
    ifconfig $VBOX up
    brctl addif vbox-br1 $VBOX
  done
fi

if test "$ACTION" = "stop" ; then

    # remove bridge
    brctl delbr vbox-br1

fi
```

QuickStart & Debug

Sample config and debug explanation are locate on a dedicated page [here]

Bugs/Limits

Parser is bash based and very primitive.It is very sensitive to "strings", especially labels/names of rules must be in pure basic alphanumeric characters. This is because parser build a shell variable with each label, which explain why label are so restrictive. As a result "MyZone or MY_Zone" is OK when "My-Zone or My:Zone" is not.

When using shared shell variables (ex: for PUBLIC-IP, Virtual-Interface names, ...) make sure that your "1st-defintion.conf" configuration file owning those variables is readed before the config using it *[*.conf are readed is alphabetic order]* In case of doubt use *"Fridu-firewall start dummy=1 verbose=1 dump=/tmp/iptables"* verify that the file owning shared variables definition is readed 1st, and if needed "vi */tmp/iptables" to verify that variables where effectivly expended.*

Download installation

FW-firewall is written on bash shell, and should work out of box on any Linux distribution. After download the easiest way of to copy a sample config that is close of yours, to copy it in a private directory and then to customize it to your need. This behind done, You can install VM-firewall and select your own rules as default config.

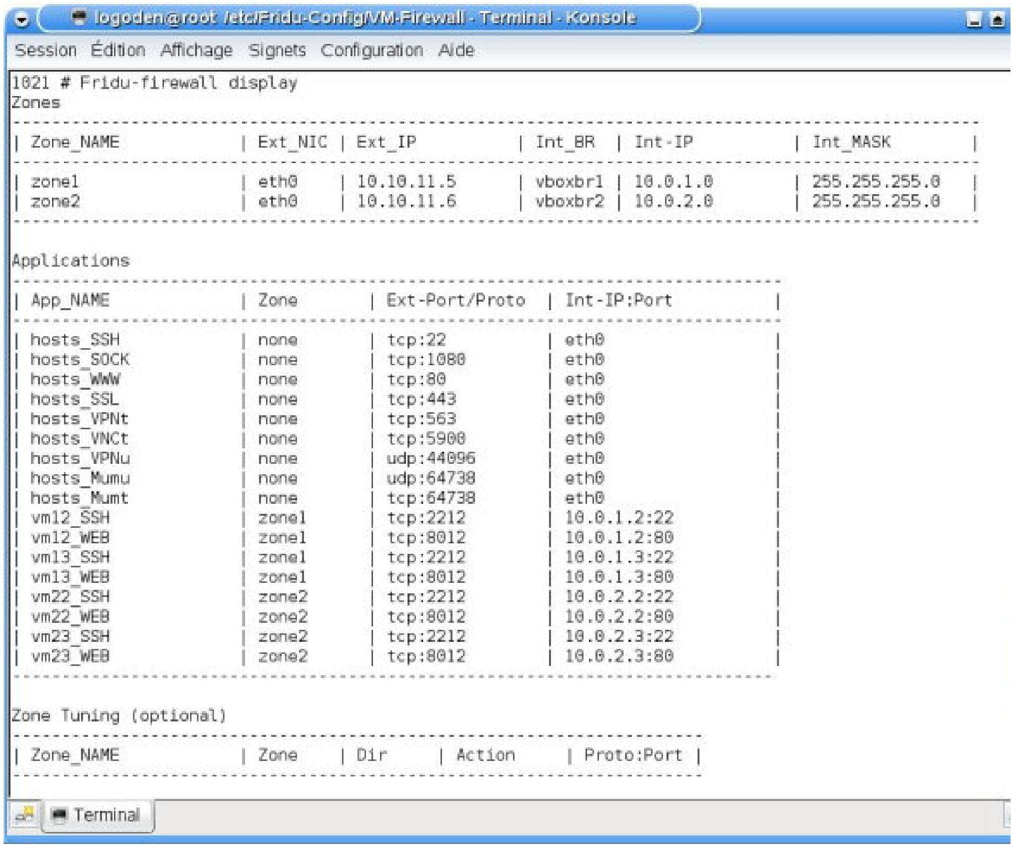
1. Downloaded VM-firewall from (here)
2. (cd /opt; tar -xzf VM-Firewall-*.tgz)

3. cd /opt/VM-Firewall
4. cp ./fw-rules/ChoosenTemplate ./fw-rules/my-$HOSTNAME
5. ./install template=my-$HOSTNAME
6. edit config in ./fw-rules/my-$HOSTNAME
7. Fridu-firestart display
8. Fridu-firewall test timeout=180
9. Fridu-firewall start

Note:

installation does not copy files, but link distribution to well know location *(/usr/sbin for commands and /etc/default | /etc/sysconfig for configurations)*. As a result "/opt/VM-Firewall" should not be deleted after installation, and building your config in /opt/FW-firewall/fw-rules or in /etc/sysconfig/Fridu-firewall is equivalent.
Autoboot: a startup script is placed in /etc/init.d and activate firewall at boot time. Two init.rc templates are provided one for Debian, the other one for SLE.
Default config: is defined by FWCONFIG=xxxx in either /etc/default/Fridu-default (debian,ubuntu) or /etc/sysconfig/Fridu-default (opensuse, redhat, ...). FWCONFIG should point *on a* template directory inside *"fw-rules"* directory. You can overload default value with *"config=xxx"* option, nevertheless init.d only start default configuration. Note that install.sh will set default template, if specifyed at installation time. To control/verify your default config use: *"Fridu.firewall display"*

Example of display command


logoden@root /etc/Fridu-Config/VM-Firewall - Terminal - Konsole
Session Édition Affichage Signets Configuration Aide
Terminal

```
1021 # Fridu-firewall display
Zones
-----------------------------------------------------------------------------------------------------
| Zone_NAME            | Ext_NIC | Ext_IP            | Int_BR  | Int-IP            | Int_MASK        |
-----------------------------------------------------------------------------------------------------
| zone1                | eth0    | 10.10.11.5        | vboxbr1 | 10.0.1.0          | 255.255.255.0   |
| zone2                | eth0    | 10.10.11.6        | vboxbr2 | 10.0.2.0          | 255.255.255.0   |
-----------------------------------------------------------------------------------------------------

Applications
-------------------------------------------------------------------------------
| App_NAME             | Zone        | Ext-Port/Proto   | Int-IP:Port          |
-------------------------------------------------------------------------------
| hosts_SSH            | none        | tcp:22           | eth0                 |
| hosts_SOCK           | none        | tcp:1080         | eth0                 |
| hosts_WWW            | none        | tcp:80           | eth0                 |
| hosts_SSL            | none        | tcp:443          | eth0                 |
| hosts_VPNt           | none        | tcp:563          | eth0                 |
| hosts_VNCt           | none        | tcp:5900         | eth0                 |
| hosts_VPNu           | none        | udp:44096        | eth0                 |
| hosts_Mumu           | none        | udp:64738        | eth0                 |
| hosts_Mumt           | none        | tcp:64738        | eth0                 |
| vm12_SSH             | zone1       | tcp:2212         | 10.0.1.2:22          |
| vm12_WEB             | zone1       | tcp:8012         | 10.0.1.2:80          |
| vm13_SSH             | zone1       | tcp:2212         | 10.0.1.3:22          |
| vm13_WEB             | zone1       | tcp:8012         | 10.0.1.3:80          |
| vm22_SSH             | zone2       | tcp:2212         | 10.0.2.2:22          |
| vm22_WEB             | zone2       | tcp:8012         | 10.0.2.2:80          |
| vm23_SSH             | zone2       | tcp:2212         | 10.0.2.3:22          |
| vm23_WEB             | zone2       | tcp:8012         | 10.0.2.3:80          |
-------------------------------------------------------------------------------

Zone Tuning (optional)
-----------------------------------------------------------------------
| Zone_NAME            | Zone     | Dir      | Action     | Proto:Port |
-----------------------------------------------------------------------
```

-

< Prev Next >

Comments (6)

generated iptables filter rules **6.** Tuesday, 23 December 2008 01:50
(Murat)

Dear Fulup,
First of all thanks for the great work! I enjoyed studying and learning from your VM-firewall framework.
I've setup VM-firewall and it works great. I've a question about the generated filtering rules. Look at these rules:
iptables -A INPUT -p icmp -j ACCEPT
iptables -A INPUT -p tcp -j REJECT --reject-with tcp-reset
iptables -A INPUT -j REJECT --reject-with icmp-port-unreachable
iptables -A OUTPUT -j ACCEPT
iptables -A OUTPUT -p tcp -j REJECT --reject-with tcp-reset
iptables -A OUTPUT -j REJECT --reject-with icmp-port-unreachable
Those are generated by VM-firewall. Can you please explain those six lines to me? (they are rendered at the end of the input/output chain)
Kind regards,Murat, The Netherlands
============== Fulup Answer ==================
Those are anti-scan rules, you should find something equivalent on most firewalls, they help fighting basic attacks by limiting your internet visibility to port scanning. First set (--reject-with icmp-port-unreachable) prevent returning a status on close ports, Fridu-firewall just refuses requests. Second one (--reject-with tcp-reset) prevents from returning an "open but protected" status. If necessary you may reduce some more your visibility by adding "-m limit --icmp-type 0 --limit 5/minute" to the "icmp -J ACCEPT". At the end of the day those rules do not impact open ports, they only slow down port scanning. Try an nmap on your system with and without Fridu-firewall, you will see how slow it is to respond when those rules are activated.

Thanks, yet I have some confusions [may sound foolish ;-)]
5. Monday, 08 December 2008 12:47
(Zenny)

It worked perfect as you have suggested and in proxmox, the default installation identify the external interface as vmbr0.
# brctl show
bridge name bridge id STP enabled interfaces
vmbr0 8000.MAC_ADDRESS yes eth0
I am stuck somewhere. I have installed the Fridu-firewall (really wonderful, no words to appreciate) in HN. But I already have a few questions. I have a network like this:
Internet ---> Gateway ---> OPENVZ in HN Node ---> ISPConfig3 in one of the VEs --> A separate * voip server machine (not in VE).
I want to do virtual hosting with ISPConfig3 (http://www.howtoforge.com/forums/showthread.php?t=26988).
But the problem I am encountering is I have same ports running in VEs (like 80. 81, 443, 53 and so on) as well as in the external voip server with freepbx.
How to port forward different ports that comes to the OpenVZ Proxmox-Fridu Firewall machine in such a case to different machines?
Second, I would like to host my own DNS server (which already is installed in ISPConfig3 VE, in this case it is MyDNS). How can I assign two public IPs to run two different instances of DNS server in the same machine? I am a bit confused, with the IP_TWO statement. Where do I need to specify in proxmox installation?
Thanks!
PS: If you wish I can render some help to proof-read and copywrite some of the articles which has some typos and grammatical mistakes. Let me know.

============ Fulup Respond =============

Good to see that now its working :)

1) Would love some help on typo, spelling and other English errors, please contact me by mail http://www.fridu.org/contact-fulup
2) I've some issue to understand your network architecture. Your alternate VOIP server must be connected either at the gateway level or through a second NIC from your HN. In first case Fridu-firewall cannot provide any help, in second case your external machine can be view as a VE from Fridu-firewall point of view.
3) IP_TWO and multiple zone, is only if you have more than one public IP pointing to your HN (on fridu I've two public IPs, but your may get anything on between 1 and more than 16 depending on your hosting plan)
4) For Virtual hosting, you cannot leverage it with Firewall port-forwarding, this is the reason why on Fridu I do use pound reverse proxy to handle virtual hosting. In fact it is theoretically possible, but in the real world, it is a much better idea to use a reverse proxy. This being said you can either run the reverse proxy on VE or HN. Nevertheless if you have more than one PUBLIC-IP available you can still use virtual hosting for each given public-ip.
5) While running to independent instances of DNS server using OpenVZ+VM-firewall is possible, it is nevertheless a bad idea. The reason for imposing two DNS is failover support, if you chose to run both on the same hardware box, any major error will kill both of them :(This being said in order to nevertheless make it, you only need to define two zones (one per public-ip) and attache one guest to each of them (TIP: do not forget than DNS use UDP for request but TCP for zones transfert)

All except port 22 is not accessible!! **4.** Friday, 05 December 2008 14:05
(Zenny)

First, thanks for the great tutorials and firewall-script. I installed the firewall and I can access the port22 of the container from outside but not other ports like 80, 443. I tried both with tuning.conf and without. In both cases the results are the same.
My configurtaions:

```
# zones.conf
CreateZone NAME=zOne NIC=vmbr0 EXT=$IP_ONE BR=venet0 INT=192.168.9.156 MASK=255.255.255.0
1st-common.conf
IP_ONE=public IP
# hypervisor.conf
CreateApp NAME=SSH ZONE=none EXT=tcp:22 INT=vmbr0
CreateApp NAME=WWW ZONE=none EXT=tcp:80 INT=vmbr0
CreateApp NAME=SSL ZONE=none EXT=tcp:443 INT=vmbr0
# vz-openssh.conf
CreateApp NAME=SSH ZONE=zOne EXT=tcp:2215 INT=192.168.9.156:22
CreateApp NAME=WWW ZONE=zOne EXT=tcp:8015 INT=192.168.9.156:80
CreateApp NAME=TOM ZONE=zOne EXT=tcp:8815 INT=192.168.9.156:8180
```

I could not figure out where did I go wrong?

=========== Fulup response ===================================
1) It is normal that tuning.conf does not change anything it is only necessary for complex configuration (ex:VPN)
2) CreateZone INT=192.168.9.156 should be INT=192.168.9.0 INT=xx.xx.xx.xx is a network mask not an IP address.
3) CreateZone NIC=vmbr0 is probably wrong because "vmbr0" is not your Ethernet interface!!! Looking to the name is must be a network bridge. Check this with "brctl show", when this bridge will be removed you will probably come back to something more traditional like "NIC=eth0"

Note: Looks like you're trying to used "veth" and not "venet". With VM-firewall Open-VZ user should always select "venet". Even if it technically possible to use VM-firewall+veth, outside very special cases like benchmarking/education to compare both venet/veth technologies,Open-VZ+VM-firewall users MUST chose venet (http://wiki.openvz.org/Venet)

VM- Firewall script closed all my Proxmox ports **3.** Tuesday, 02 December 2008 03:08

(Trevor Williams)

I am testing the firewall script on my proxmox node, namp shows all my ports locked after starting the firewall script. How could this be. Should I do a re-install. Help me... I don't have funds to buy public IPs' so getting to this to work would be great... thanks

```
MY CONFIG
IP_ONE=10.10.10.12
IP_TWO=192.168.1.40
CreateZone NAME=zOne NIC=eth0 EXT=$IP_ONE BR=venet0 INT=10.10.101.0
MASK=255.255.255.0
CreateZone NAME=zTwo NIC=eth0 EXT=$IP_TWO BR=venet0 INT=10.10.102.0
MASK=255.255.255.0
# Hyperviser Application Port Forwarding
CreateApp NAME=DOM0_SSH ZONE=none EXT=tcp:22 INT=eth0
CreateApp NAME=DOM0_WWW ZONE=none EXT=tcp:80 INT=eth0
CreateApp NAME=DOM0_SSL ZONE=none EXT=tcp:443 INT=eth0
CreateApp NAME=DOM0_VPNt ZONE=none EXT=tcp:563 INT=eth0
# Zone one Application ports Forwarding
CreateApp NAME=Mail_SMTP ZONE=zOne EXT=tcp:25 INT=10.10.101.1:2525
CreateApp NAME=Mail_IMAP ZONE=zOne EXT=tcp:993 INT=10.10.101.1:993
CreateApp NAME=Domi_WEB ZONE=zTwo EXT=tcp:80 INT=10.10.102.2:80
CreateApp NAME=Domi_SSH ZONE=zTwo EXT=tcp:22 INT=10.10.102.2:22
CreateApp NAME=Bren_SSH ZONE=zOne EXT=tcp:2216 INT=10.10.101.5:22
CreateApp NAME=Bren_WEB ZONE=zOne EXT=tcp:8016 INT=10.10.101.5:80
```

======== Fulup Respond ======================
GOOD NEWS: what you're trying to acheive is working out of the box :)
BAD NEWS: you took config sample without touching it, and it would be strange that it fit your environment :(
==
This being said: all external port cannot be lock with this config. You open port 22/SSH with zone=none in which case this port will be open even if your external IP address was wrong.

Advice:
- simplify your config,a nd move to multi-file model as in fw-rules/openvz
- triple check your public-IP (both your IP_ONE and IP_TWO are not routable !!!)
- make sure your VM are reachable at 10.10.10x.xyz
- verify your config file is in fw-rules/default[.conf] or force config=xxxx in command line

Verification
- Fridu-firewall display ;# verify you activate the config file you want
- Fridu-firewall test dump=/tmp/iptables.dump ;# check your low level rules
- iptables -L | grep ssh ; # should see "anywhere tcp dpt:ssh"
- Never forget to do your nmap check from Internet, cannot work from hypervisor ;# typical error

Outside if this it's guaranty to work :)

Port forwarding works **2.** Sunday, 23 November 2008 13:16

(Adam Ryczkowski)

I've managed to put the port forwarding to work. Actualy it was working all the time.
It's shame to admit, but the problem was my (other) firewall, which blocked access to the openvz from hardware node to the only other host I was testing the access. This fact together whith the fact, that I was unable to tap to the guest system from the hardware node itself via it's _external_ interface (which as I understand now is a normal behiavour) pushed me to write the previous letter.

Thank you very much for publishing this simple firewall script as well as all other howtos!
Adam

==== Fulup respond ==== I love problems, when they solve by themselves :)

Port forwarding doesn't work **1.** Saturday, 22 November 2008 13:18
(Adam Ryczkowski)

I'm trying to implement openvz cluster with the help of your firewall script. I managed to have internet access on the first try from the guest system, but I still have problem with any port forwarding. Regardless of what I put in the configuration file, the forwarded port is closed (state: filtered).

I've set up openvz on ubuntu server 8.04, with your Fridu-firewall. The host has only one nic "eth0" with static IP 192.168.3.46. Yes, it's private IP, but in my testing environment it might be considered public. The guest has IP 192.168.11.200 and ssh deamon up and running correctly.

Here goes my minimalistic configuration:

1st-common.conf:

```
IP_ONE=192.168.3.xx
```

zones.conf

```
CreateZone NAME=zOne NIC=eth0 EXT=$IP_ONE BR=venet0 INT=192.168.11.0 MASK=255.255.255.0
```

hypervisor.conf:

```
CreateApp NAME=SSH ZONE=none EXT=tcp:22 INT=eth0
```

tuning.conf: (I left almost everything untouched, as I don't really understand what it does)

```
DoIt iptables -A after-forwarding -i venet+ -o venet+ -j ACCEPT
DoIt iptables -A after-input -i tun+ -j ACCEPT
DoIt iptables -A after-forwarding -i tun+ -o venet+ -j ACCEPT
DoIt iptables -A after-forwarding -i venet+ -o tun+ -j ACCEPT
```

I've commented out the last line, as i do not wish to give two "public" IPs.

vz.conf: I wish to forward internal port 22 to external 2022

```
CreateApp NAME=SSH ZONE=zOne EXT=tcp:2022 INT=192.168.11.200:22
```

Now, the guest has full access to the internet. I'm able to connect to the ssh deamon from host, if I type "ssh 192.168.11.200" (and of course get "open" port report with "nmap 192.168.11.200 -p 22 "). Typing "ssh 192.168.3.xx -p 2022" doesn't work on any computer on 192.168.3.0/24 subnet, nor "nmap 192.168.3.xx -p 2022" doesn't show open port.

===== Fulup Response =====
*** you do not need "tuning.conf" this file is only necessary when running multiple zones and/or a VPN (just delete it).
*** the rest of your config is fine. The two most typical errors you may have are:
1) you have an other firewall that locks port 22 on your guest (verify packets reach your guest "tcpdump -i venet0 port 22" on your guest)
2) you try to access service directly from the hypervisor (ssh -p 2022 192.168.3.46 will only work from internet)
3) your infrastructure (router, modem, operator, ...) lock port 2022
--- Outside of that it should work :)

Add your comment

Your name:

Your email:

Subject: