

# vPF\_RING User's Guide

High Speed Packet Capture On Virtual Machines

Version 1.0.0  
July 2011

© 2011 ntop.org

# 1. Table of Contents

Introduction .....	3
What's New with vPF_RING User's Guide? .....	3
Terminology .....	3
Prerequisites .....	3
License .....	3
Welcome to vPF_RING .....	4
vPF_RING Installation .....	5
Installation Prerequisites (Host side) .....	5
Patched QEMU Installation (Host side) .....	5
Demo Virtual Machine .....	6
Running the Virtual Machine .....	6
vPF_RING SDK .....	7
vNPlug Kernel Module Installation (Guest side) .....	7
vPF_RING Library Installation .....	7
Using vPF_RING .....	8
Checking PF_RING Device Configuration .....	8
Libpfring and Libpcap .....	8

## 2. Introduction

vPF\_RING is a high speed packet capture framework that turns a Virtual Machine running on a commodity PC into an efficient network measurement box.

### 2.1. What's New with vPF\_RING User's Guide?

- Release 1.0 (July 2011)
  - Initial vPF\_RING users guide.

### 2.2. Terminology

Throughout this document we use the following terms:

- Guest  
This term indicates the virtual machine operating system. In other words this is the virtual machine that is running the virtualized Linux environment.
- Host  
The host is the physical machine (bare hardware) on which the hypervisor runs. Virtual machines are sitting on top of the hypervisor.

### 2.3. Prerequisites

Below you can find the list of main vPF\_RING prerequisites:

- Linux kernel 2.6.30 or better, with KVM support.
- 64 bit host Linux.
- OS guests supported by vPF\_RING are limited to Linux.
- Hardware system with virtualization support enabled in the BIOS (required by KVM).

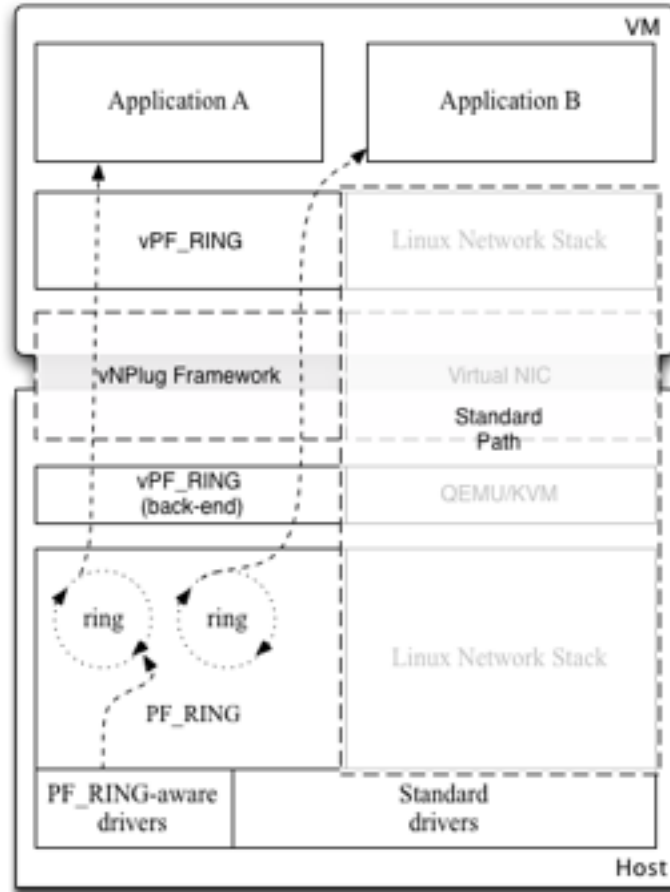
### 2.4. License

All vPF\_RING components are distributed in source format and stored inside <PF\_RING>/vPF\_RING.

In order to develop applications on top of vPF\_RING, it is necessary to get the vPF\_RING SDK. Please refer to chapter 5.

### 3. Welcome to vPF\_RING

vPF\_RING's architecture is depicted in the figure below.



The main building blocks from the bottom are:

- Specialized PF\_RING-aware drivers (optional) (host side) that allow to further enhance packet capture by efficiently copying packets from the driver to PF\_RING without passing through the kernel data structures. For further information please refer to the PF\_RING User's Guide.
- The standard PF\_RING kernel module (host side).
- The vPF\_RING backend (host side), that interacts with the standard PF\_RING module on the host side, and the user-space library on the guest side by means of the vNPlug Framework.
- The vNPlug Framework (host and guest side), that provides a direct mapping of the PF\_RING memory structures on the guest and a reliable communication channel. This framework comes as a QEMU patch on the host side, and a kernel module on the guest side.
- The user-space vPF\_RING library that provides transparent PF\_RING-support to user-space applications on the VM.

Incoming packets are copied by the kernel module on the host side into a memory ring allocated at creation time, and directly read by the user-space applications on the VM.

Applications can issue standard PF\_RING API calls, described in the PF\_RING User's Guide.

## 4. vPF\_RING Installation

vPF\_RING source code is distributed with PF\_RING. Download vPF\_RING as explained in [http://www.ntop.org/products/pf\\_ring/vpf\\_ring](http://www.ntop.org/products/pf_ring/vpf_ring)

The <PF\_RING>/vPF\_RING source code layout is the following:

- README
- docs/
- guest/
- host/
- img/

### 1.Installation Prerequisites (Host side)

The vPF\_RING installation expects that PF\_RING is compiled and installed on the host system.

```
host $ cd <PF_RING>/kernel
host $ make
(as root do)
host # make install
host # insmod pf_ring.ko
```

```
host $ cd <PF_RING>/userland/lib
host $ ./configure
host $ make
(as root do)
host # make install
```

Note: if you want to use a PF\_RING-aware drivers with `transparent_mode` or other settings, please refer to the PF\_RING User's Guide.

### 2.Patched QEMU Installation (Host side)

Compile and install the patched QEMU (part of this distribution) as explained below:

```
host $ cd <PF_RING>/vPF_RING/host
host $ ./configure
host $ make
(as root do)
host # make install
```

Note that you cannot use the QEMU binary that comes with your distribution, as we need to patch it in order to support vPF\_RING.

## 5. Demo Virtual Machine

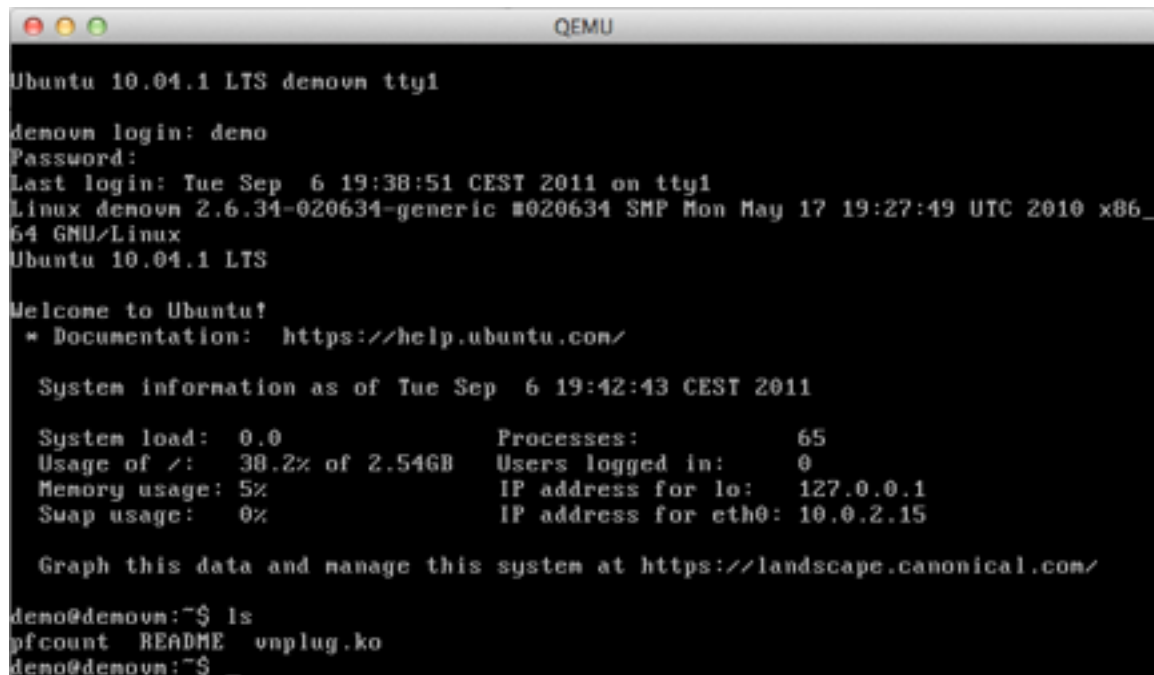
For your convenience we provide you a simple demo image you can find inside <PF\_RING>/vPF\_RING/img. The default user of the demo image is 'demo' with password 'VPF\_RING'. Once you are inside the VM, you can 'sudo su' to switch to root.

### 5.1. Running the Virtual Machine

In order to run the VM with vNPlug/vPF\_RING support use the "-device vnpug" parameter that instructs QEMU to enable vPF\_RING support. In order to run a guest you need to supply as parameter a virtual machine disk image.

```
host # modprobe kvm_intel
host # /usr/local/kvm/bin/qemu-system-x86_64 \
    -hda ubuntu-amd64.img \
    -boot c \
    -m 512 \
    -vnc 0.0.0.0:0 \
    -device vnpug
```

After you started your VM, you can connect to it using a VNC client that will connect to the IP address of the server running QEMU. Optionally you can configure a virtual network interface to ssh to it. Please refer to KVM guide for more information, or use the utility scripts present in <PF\_RING>/vPF\_RING/img.



```
QEMU
Ubuntu 10.04.1 LTS demovn tty1
demovn login: demo
Password:
Last login: Tue Sep  6 19:38:51 CEST 2011 on tty1
Linux demovn 2.6.34-020634-generic #020634 SMP Mon May 17 19:27:49 UTC 2010 x86_64 GNU/Linux
Ubuntu 10.04.1 LTS

Welcome to Ubuntu!
 * Documentation:  https://help.ubuntu.com/

System information as of Tue Sep  6 19:42:43 CEST 2011

System load:  0.0               Processes:            65
Usage of /:   38.2% of 2.54GB    Users logged in:     0
Memory usage: 5%               IP address for lo:    127.0.0.1
Swap usage:   0%               IP address for eth0:  10.0.2.15

Graph this data and manage this system at https://landscape.canonical.com/

demo@demovn:~$ ls
pfcount  README  vnpug.ko
demo@demovn:~$
```

Note: when using the preconfigured VM, please refer to the README file present in the VM image inside the home directory.

## 6. vPF\_RING SDK

### 1.vNPlug Kernel Module Installation (Guest side)

In order for the vNPlug framework to work properly, you should load the acpihp kernel module (hotplug support), otherwise it won't be able to dynamically map ring memory.

```
(as root do)
guest # modprobe acpihp
```

Compile and install the vNPlug kernel module:

```
guest $ cd <PF_RING>/vPF_RING/guest/kernel
guest $ make
(as root do)
guest # make headers_install
guest # insmod vnplug.ko
```

### 2.vPF\_RING Library Installation (Guest side)

Compile and install the PF\_RING library with vPF\_RING support:

```
guest $ cd <PF_RING>/userland/lib
guest $ ./configure
guest $ make
(as root do)
guest # make install
```

Example: compile and run pfcount:

```
guest $ cd <PF_RING>/userland/examples
guest $ make pfcount
(as root do)
guest # ./pfcount -i host:eth0
```

Note: as you can read from [https://svn.ntop.org/svn/ntop/trunk/PF\\_RING/vPF\\_RING/README.LICENSE](https://svn.ntop.org/svn/ntop/trunk/PF_RING/vPF_RING/README.LICENSE), our work is self-funded and we need some income in order to continue with our research, thus we decided to ask a little fee for using the vPF\_RING library. See <http://shop.ntop.org> to get a per-host unlock code. With no unlock-code the application can work for a few minutes in order to allow you to evaluate it.

## 7.Using vPF\_RING

Before using any PF\_RING application the pf\_ring kernel module should be loaded on the host side.

```
host # insmod <PF_RING>/kernel/pf_ring.ko
```

Note: if you want to use a PF\_RING-aware drivers with transparent\_mode or other settings, please refer to the PF\_RING User's Guide.

On the guest side both the standard hotplug module and the vnplug module should be loaded.

```
guest # modprobe acpiphp  
guest # insmod vnplug.ko
```

### 7.1. Checking PF\_RING Device Configuration

As with standard PF\_RING, when a ring is activated a new entry /proc/net/pf\_ring is created on the host.

```
host # cat /proc/net/pf_ring/info  
Version      : 4.7.1  
Ring slots   : 4096  
Slot version  : 13  
Capture TX   : Yes [RX+TX]  
IP Defragment : No  
Socket Mode  : Standard  
Transparent mode : Yes (mode 0)  
Total rings   : 0  
Total plugins : 2
```

### 7.2. Libpfiring and Libpcap

As vPF\_RING results in a standard PF\_RING module which is hidden by the PF\_RING API, both libpfiring and libpcap can be compiled and used as described in the PF\_RING User's Guide.

Note: in order to indicate to the library to use the vPF\_RING module, you need to prepend 'host:' to the device name (e.g. host:ethX@Y).