PSA Creating and Testing

[1 Overview 2](#_Toc399937634)

[1.1 Technical requirements 2](#_Toc399937635)

[1.2 Testbed layout 2](#_Toc399937636)

[2 Testbed setup 4](#_Toc399937637)

[2.1 Creation of virtual isolated network 4](#_Toc399937638)

[2.2 Generate the two traffic machines 4](#_Toc399937639)

[2.3 PSA creation 5](#_Toc399937640)

[3 Testbed execution 7](#_Toc399937641)

[3.1 Accessing to VM 7](#_Toc399937642)

[3.2 PSA configuration 7](#_Toc399937643)

[3.3 Test workflow 8](#_Toc399937644)

[3.4 PSA manifest file 8](#_Toc399937645)

# Overview

This document aims at presenting how to create and test a personal security application (PSA) with minimal effort, without having to deal with the full SECURED architecture.

This document is oriented to the developers of PSAs, who need to test the behavior of their software in a controlled environment.

## Technical requirements

The current SECURED solution operates in a KVM virtualized environment and the PSAs are virtual machines that receive traffic from an underlying software switch embedded in the host hypervisor, which is currently represented by the OpenvSwitch software.

As a consequence, the developer should have an host with the following software:

1. A virtualized enviroment qemu-KVM (http://www.linux-kvm.org/page/HOWTO).

2. A client with graphical enviroment with virtmanager installed, which is used to connect to the virtualized environment through the ssh protocol.

However, the test environment includes three VMs directly connected (traffic generator VM 🡪 PSA under testing 🡪 traffic receiver VM), so the software switch (OpenvSwitch) is not required for the test and creation of the PSA.

To deploy a PSA VM, the programmer should create a qemu-KVM disk image generated by libvirt on the VM creation. The disk image, in the basic configuration, will be located under the /var/lib/libvirt/images folder named with the VM name used in its creation.

## Testbed layout

The figure represents the testbed that is dedicated to test the new PSA. The two “green” virtual machine are provided to the PSA developers with an extensive set of traffic generator/receiver tools and are directly connected to the PSA under test. The traffic generator machine is connected to the data-IN interface of the PSA and the traffic receiver machine is connected on the data-OUT interface.

The two virtual machines are already configured with static IP addresses (depicted on the Figure) that permits to communicate each other.

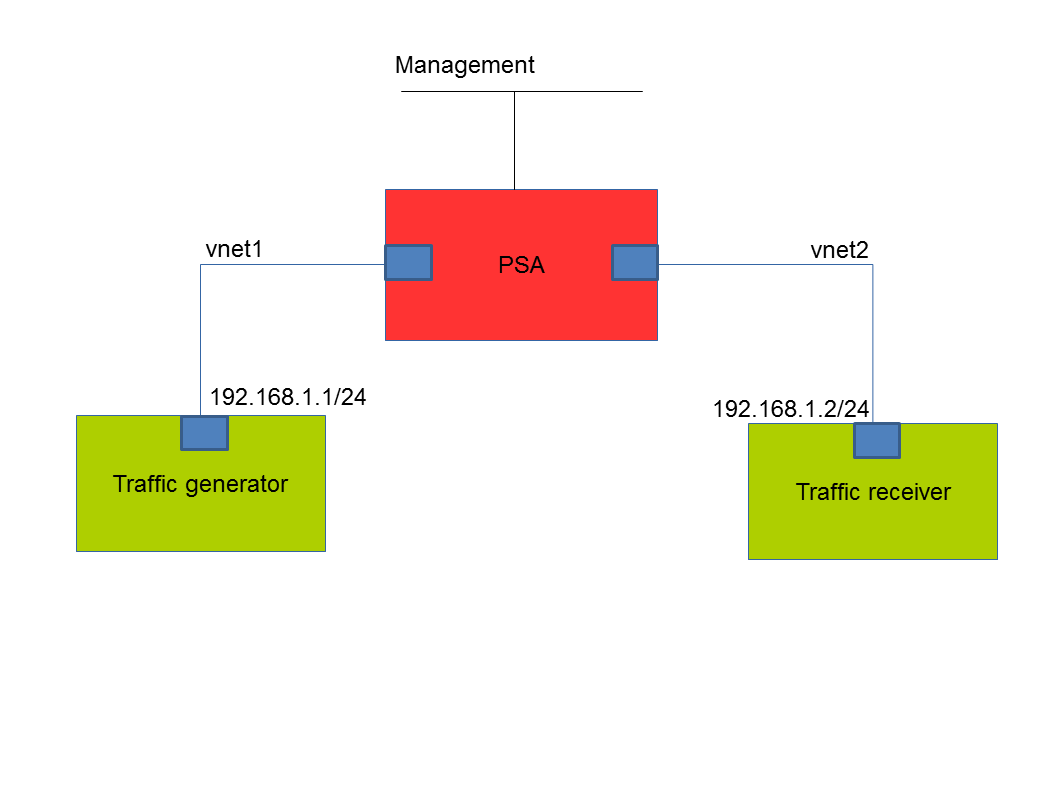


Figure 1: Test scenario

The Figure 1 depicts how the three VM should be connected. The PSA requires also another interface for the control/configuration network. Depending on the requirements of the developer, this interface may be configured in order to offer Internet connectivity to the PSA. However this is out of the scope of this document.

# Testbed setup

The complete setup of the testbed requires to carry out the following three steps:

1. Create the proper (logical) network infrastructure in order to allow the traffic to flow between our components. This step has to be completed before installing all the VMs.
2. Import and install the VM images of the traffic generator/receiver
3. Create the VM image for the PSA under testing

## Creation of virtual isolated network

From the virtual machine manager (virtmanager) interface select Edit->Connection Details and select Virtual Networks and add two “Isolated virtual network”, disabling the IP network address space definition and the DHCP. The two network will be called vnet1 and vnet2. An example of this configuration is depicted in the figure below.

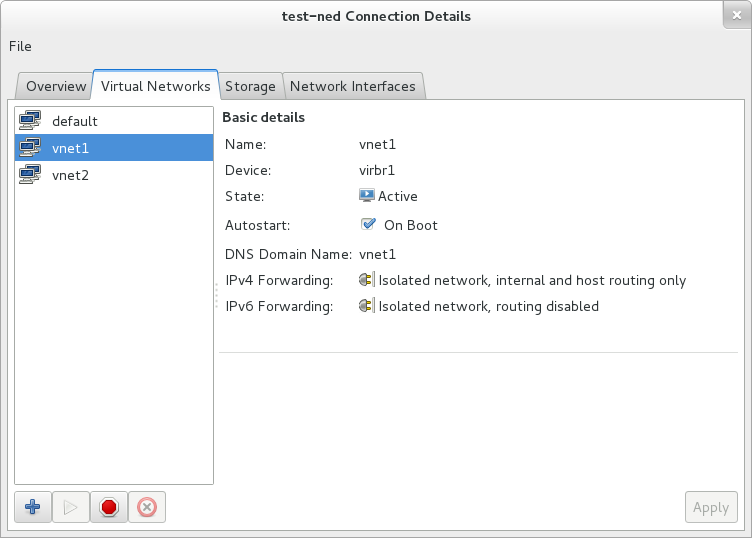


Figure 2: Virtual network creation interface

## Installing the two traffic generator/receiver machines

Although the VM images of the traffic generator/receiver machines are already available for download, the virtmanager application requires the developer first to create two (void) VMs, then update those VMs with the proper image.

Hence, we have to connect to the testing machine from the virtmanager application. This could be done either remotely through an ssh connection or in the same machine in localhost.

The two disk images of TrafficGen.img and TrafficRcv.img should be in /var/lib/libvirt/images folder on the host machine.

When creating the two virtual machines, we have to select “import existing disk image” and use the TrafficGen.img in case of the first machine and the TrafficRcv.img for the second one.

In the next step is possible to select the amount of memory and CPUs assigned to the VM (1024 MB RAM and 1 vCPU is suggested). In the last step of the creation check the “Customize configuration before install” option and on the NIC options select the Virtual network ‘vnet1’ as Source device for the first machine and the Virtual network ‘vnet2’ for the second one. In the “device model option” select “Hypervisor default”.

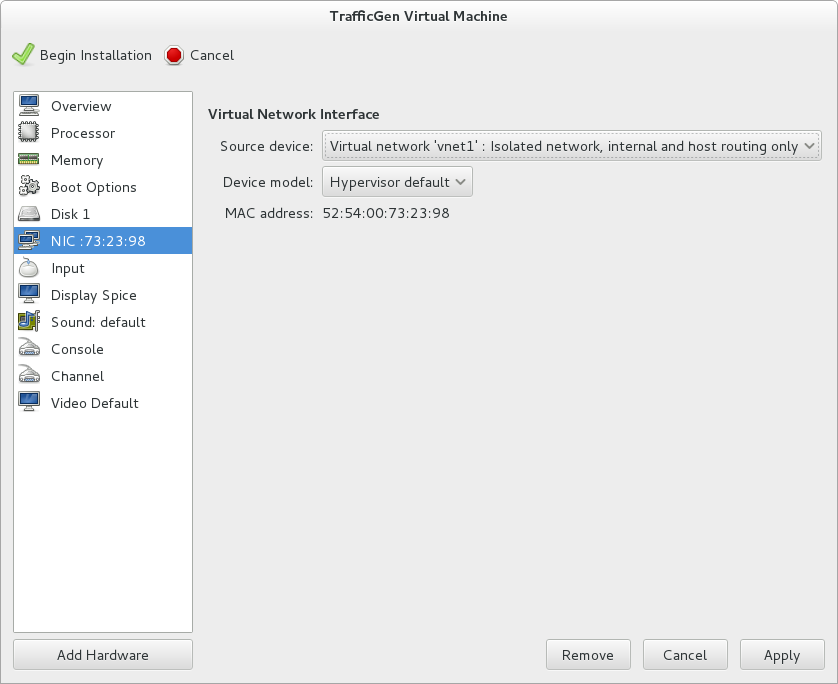


Figure 3: Connect an interface

To actually create the VM you have to select the “Begin Installation” button.

## PSA creation

To create the PSA it is possible to use an already generated disk image or create a new one with a new installation, which should be done using the specific OS installation disk (ISO file or physical drive). For the new installation it is required to create the disk image specifying the size (for the correct future work the “Allocate entire disk now” option should be checked). Before the creation check the “Customize configuration before install” box. Select Virtual network ‘vnet1’ as Virtual Network Interface and use the Add Hardware button to add another network interface selecting Virtual network ‘vnet2’ as Host device. A third virtual NIC should be added for management/configuration purposes, which should be connected in “Virtual network 'default': NAT mode”. An example of this last configuration is shown in the figure below.

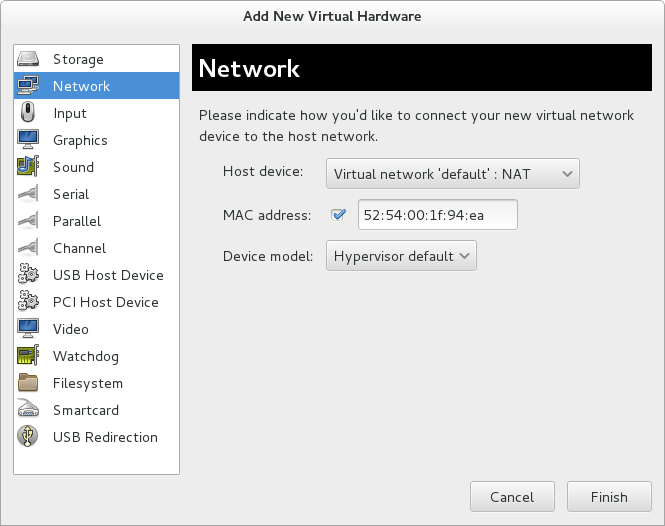


Figure 4: Add new interface

Now the configuration of the three virtual machines are the one depicted in the Figure 1 and the PSA can use the third interface to get internet connectivity for its configuration.

# Running the testbed

## Accessing to VM

First, we should connect to the created VMs through VNC protocol. This can be done by using the virtmanager graphical interface, as shown in the figure below.

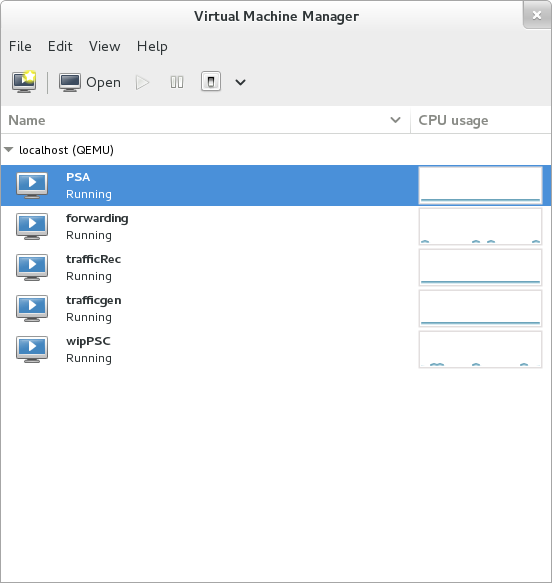


Figure 5: Virtual Machine Manager interface

## PSA configuration

For the demo setup the PSA requires a static configuration for the network interfaces:

1. Data interfaces: they are the first two interfaces, the first one for the “internal”[[1]](#footnote-1) traffic and the second one for the “external” traffic and they should work transparently forwarding the internal traffic to the outside and vice versa.
2. Management interface: it is the last one and it requires to be configured to get a dynamic IP from the DHCP server in the control network that is associated with the PSC.

## Test workflow

The PSA VM has the following credentials (also root):

user: psa

pass: secured

The two VM with traffic generator/receiver can be accessed with the following credentials:

user: traffic

pass: trafficMachine

They are configured with the following ip: 192.168.1.1/24 for the trafficGen and 192.168.1.2/24 for the trafficRec.

Both machines have installed an Apache server and can communicate with the curl tool already installed on both. To see the PSA working the two machines should communicate through the network using their configured IP and their traffic should be forwarded by the created PSA.

# Finalizing the PSA deployment

## PSA manifest

While the above procedure is enough to test the PSA in a controlled environment, the physical deployment on a NED requires one more step, which consists in creating the PSA manifest. The PSA manifest is not used in the controlled environment that we built to test the PSA. However, it will be needed by the NED in order to setup the proper virtualized environment for the PSA, such as selecting the proper virtualized system (e.g., KVM vs. VmWare), creating the right number (and types) of virtual NICs, and attach them to the rest of the system.

An example of a possible PSA manifest is shown in the picture below.

{

"PSA\_id":"12345",

"disk": "IpTraf.img",

"interface": [

{

"network":"data",

"type":"data\_in"

},

{

"network":"data",

"type":"data\_out"

},

{

"network":"control",

"type":"manage"

}

],

"memory": "1024",

"os-architecture": "x86\_64",

"vcpu": "1",

"psatype": "vm",

"exec-environment": "kvm"

}

Figure 6: example of a PSA manifest

The complete specification of the PSA manifest is available at the following address:

TBD

## PSA deployment

In the real NED, PSA images are stored in the /var/lib/libvirt/images folder. In the same folder there should be also the PSA manifest, named as follows:

TBD

1. “Internal” and “external” refer here to the typical deployment of a firewall application, which includes the “internal” site (i.e., the network that has to be protected) and the “external” side (i.e., the network where the attacks come from). [↑](#footnote-ref-1)