

How to Deploy a Virtual Test Agent Image in Virtualbox

Published
2021-01-13

Table of Contents

[Executive Summary](#)

[Paragon Active Assurance: Solution Overview](#)

[Prerequisites](#)

[Setting Up a Virtual Test Agent in VirtualBox](#)

[How to Establish Contact with Control Center](#)

[Appendix: Description of the vTA VNF and Its Requirements](#)

Executive Summary

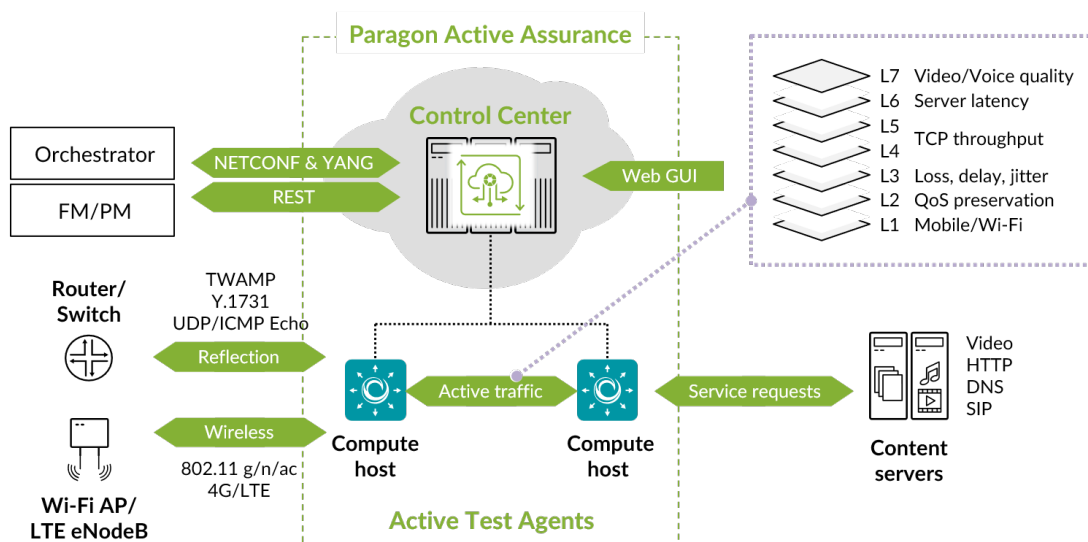
This guide explains how to start a Virtual Test Agent (vTA) from Paragon Active Assurance in VirtualBox and how to establish contact between the vTA and a Paragon Active Assurance Control Center residing outside VirtualBox.

Paragon Active Assurance: Solution Overview

Paragon Active Assurance consists of two parts:

1. **Test Agents** – software-based active traffic generators. Virtual Test Agents (vTAs) are ones that you upload and boot from your own OpenStack environment. These vTAs will automatically connect to Control Center as part of the deployment process described in this guide. (Juniper Networks also offers non-virtual Test Agents in the form of software that is installed on stand-alone x86 hardware.)
2. **Control Center** – for centralized control and coordination of Test Agents, including distributed VNF vTAs. This includes initiating test sequences and monitoring sessions, as well as evaluating collected measurement data, SLAs and KPIs.

Paragon Active Assurance vTAs are controlled through Control Center. The interface towards Control Center is either a web GUI or an orchestration API, as illustrated below:



Prerequisites

IN THIS SECTION

- [Control Center Account | 2](#)
- [vTA Image | 2](#)

Control Center Account

You need an account in a Control Center in order to access it: either the one belonging to the SaaS solution or one installed on-premise in your organization. If you do not already have a Paragon Active Assurance account, please contact your Juniper partner or your local Juniper account manager or sales representative.

vTA Image

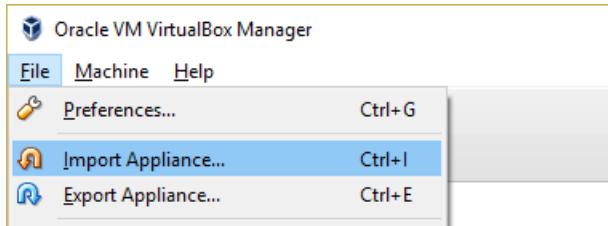
The VNF vTA image is provided either by one of Juniper's partners or directly by Juniper. If you do not already have the VNF vTA image, you can download it from the Control Center web GUI (whether SaaS or on-premise).

The vTA image is in **OVA** (OVF/VMDK) format and is packaged using the OVF Tool, which uses a SHA1 checksum. The OVF file specifies version VMX-09, since that is the lowest version which has the requisite functionality. The OVF file also specifies 512 MB RAM and 2 GB block storage for the vTA.

Setting Up a Virtual Test Agent in VirtualBox

Here is how to load the vTA from the OVA image.

- In VirtualBox Manager, select the virtual machine named "vTA".
- From the **File** menu, select **Import Appliance**.



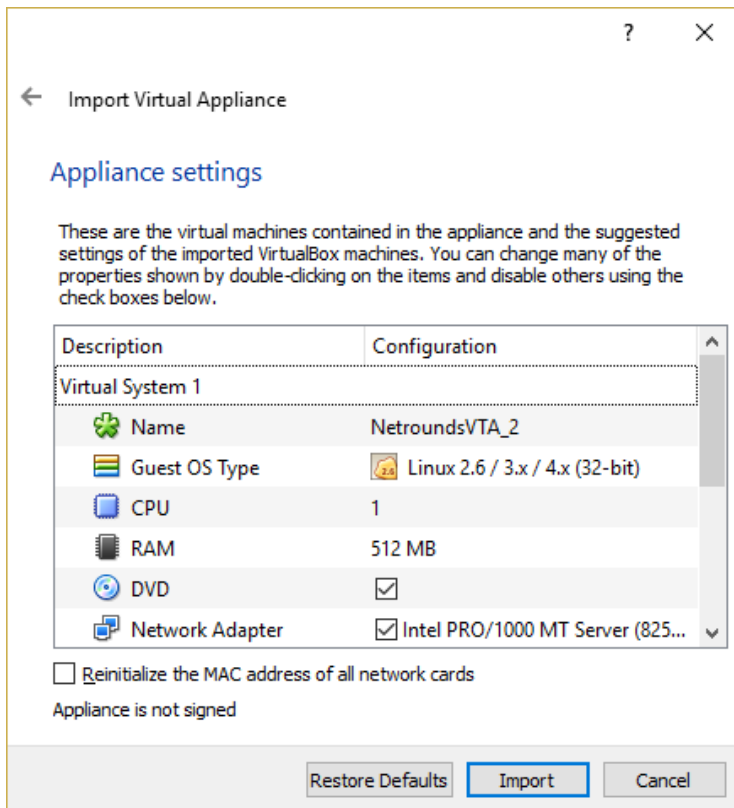
- Browse to select the OVA image file (*.ova). Then click **Next**.

Appliance to import

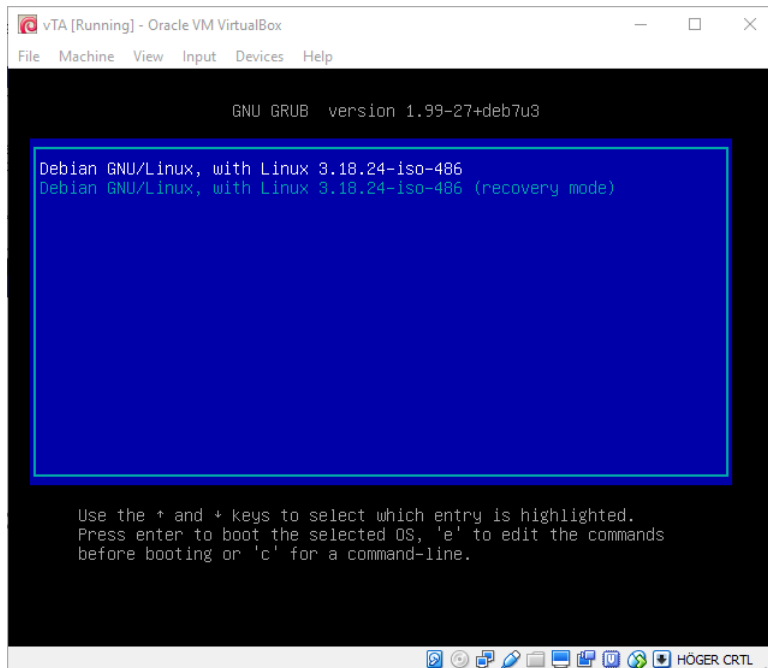
VirtualBox currently supports importing appliances saved in the Open Virtualization Format (OVF). To continue, select the file to import below.

C:\Users\...\Downloads\netrounds-test-agent-vmware_2.21.0-rc1.ova

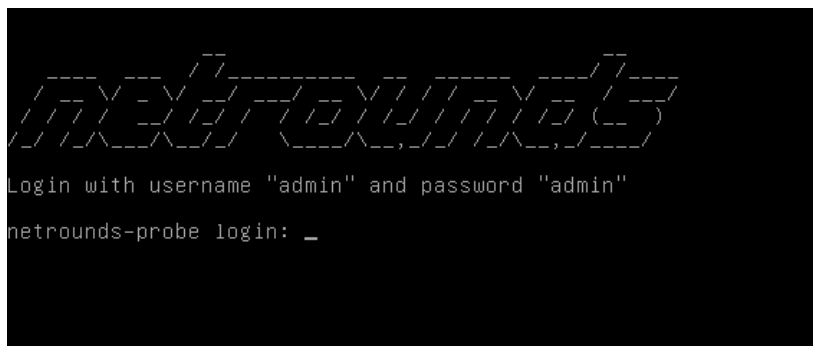
- Change the appliance settings if necessary. The default settings are however a good starting point for testing.
- Then click **Import** to go ahead with importing the OVA image.



- Now start the virtual machine named “vTA” in VirtualBox Manager. You will be prompted to log in to the virtual machine. Normally the topmost item should be selected in the screenshot that follows:



- Log in to the Test Agent using the credentials “admin”/”admin” as indicated below.



The Test Agent local console menu now appears. Proceed to register the Test Agent with a Control Center, as described in the Paragon Active Assurance in-app help under **Test Agents** → **Configuring Test Agents from the local console**.

How to Establish Contact with Control Center

IN THIS SECTION

- [Setting Up a VirtualBox Host-only Ethernet Adapter | 6](#)
- [Connecting vTAs to a Host-only Ethernet Adapter | 9](#)
- [Connecting vTAs to an Internal Network Ethernet Adapter | 10](#)

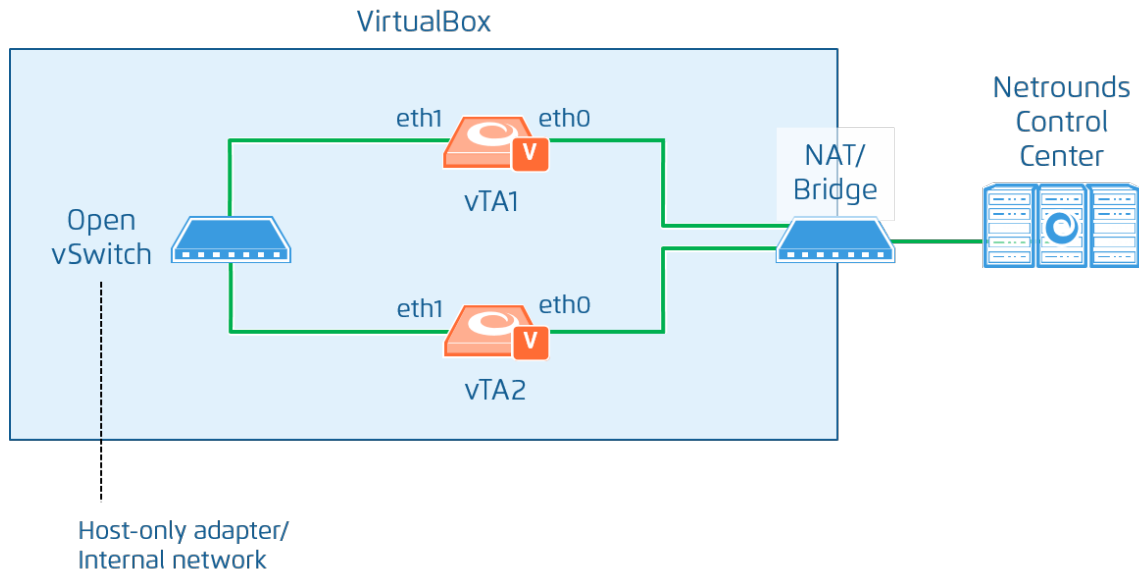
This chapter explains how to establish contact between Virtual Test Agents deployed in VirtualBox and a Control Center residing outside VirtualBox.

For communication outside VirtualBox, either NAT or a bridge can be used. In the following, the use of NAT is assumed. A bridge may however be preferable if a more transparent setup is needed: for example, to permit communication in both directions between vTAs in VirtualBox and other Test Agents installed elsewhere.

Internally in VirtualBox, the following setup is recommended for each vTA:

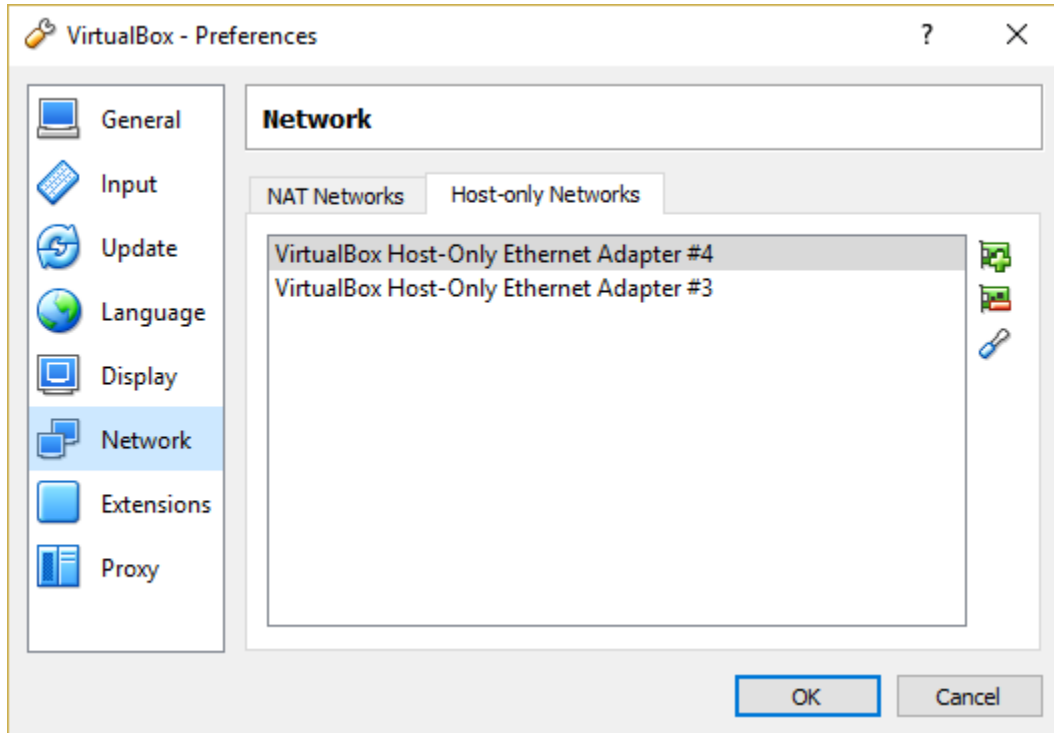
- The vTA has one network adapter (labeled “eth0” in the diagram below) attached to NAT for communication with Control Center. This is the default setting for the network adapter which is predefined for the vTA.
- The vTA has another network adapter (labeled “eth1” in the diagram) attached to a VirtualBox host-only Ethernet adapter.^[1] This connection is used for communication between vTAs in the course of testing. How to configure this is covered in the sections ["Setting Up a VirtualBox Host-only Ethernet Adapter" on page 6](#) and ["Connecting vTAs to a Host-only Ethernet Adapter" on page 9](#).


An overview of the setup is given in the diagram below.

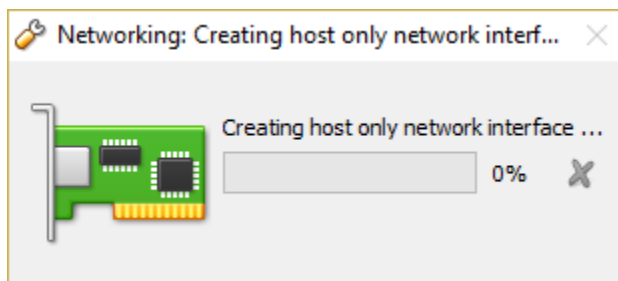


Setting Up a VirtualBox Host-only Ethernet Adapter

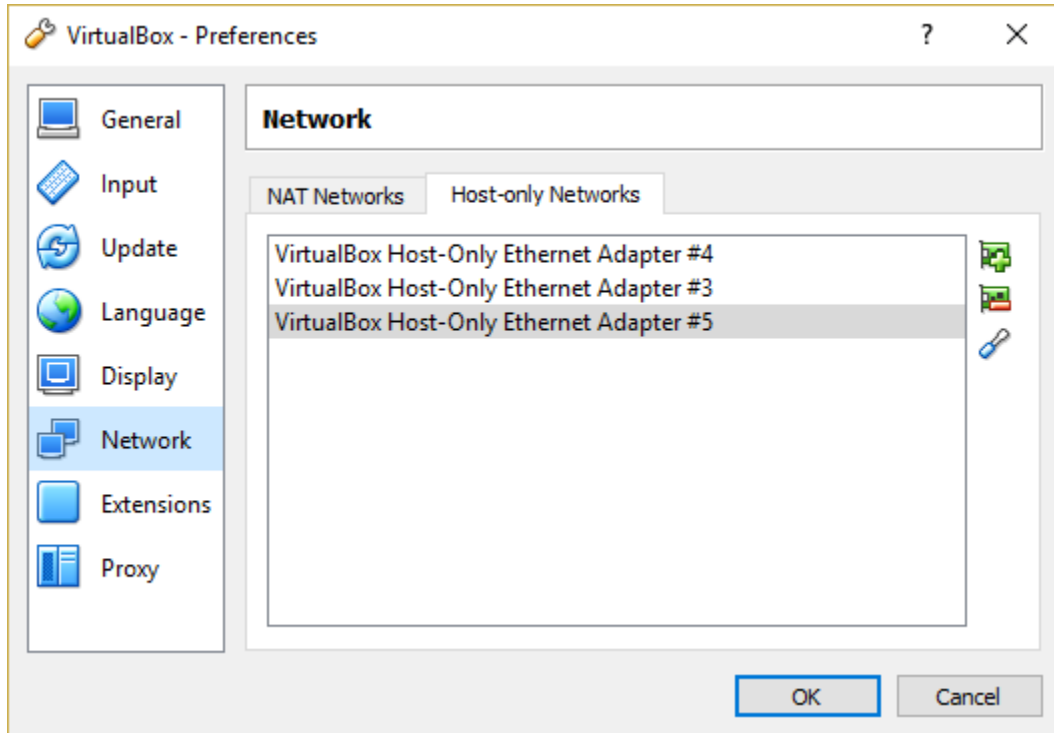
- From the **File** menu, select **Preferences**.
- Select **Network**, and click the **Host-only Networks** tab.



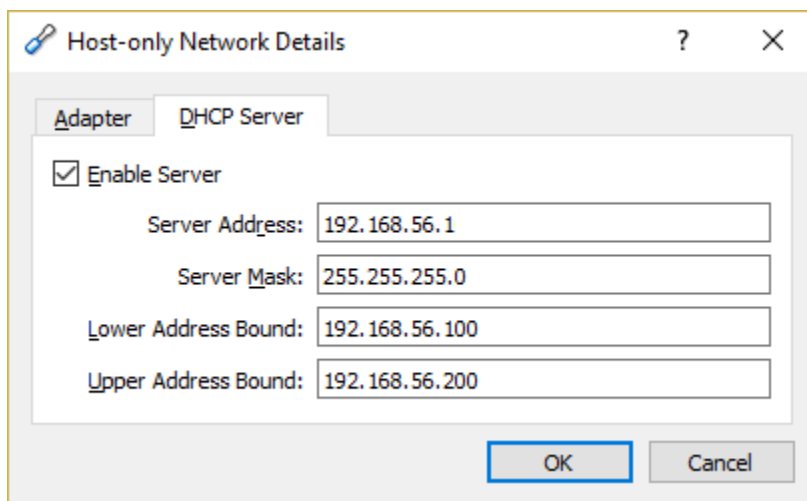
- Click the  ("New") button.
- Wait for this procedure to finish:



- A new VirtualBox host-only Ethernet adapter will now appear in the listing:



- Double-click the new Ethernet adapter.
- In the dialog that appears, select the **DHCP Server** tab.
- Check the **Enable Server** box.
- Enter **Server Address** and **Server Mask**, and set lower and upper address bounds. An example is shown in the screenshot below.

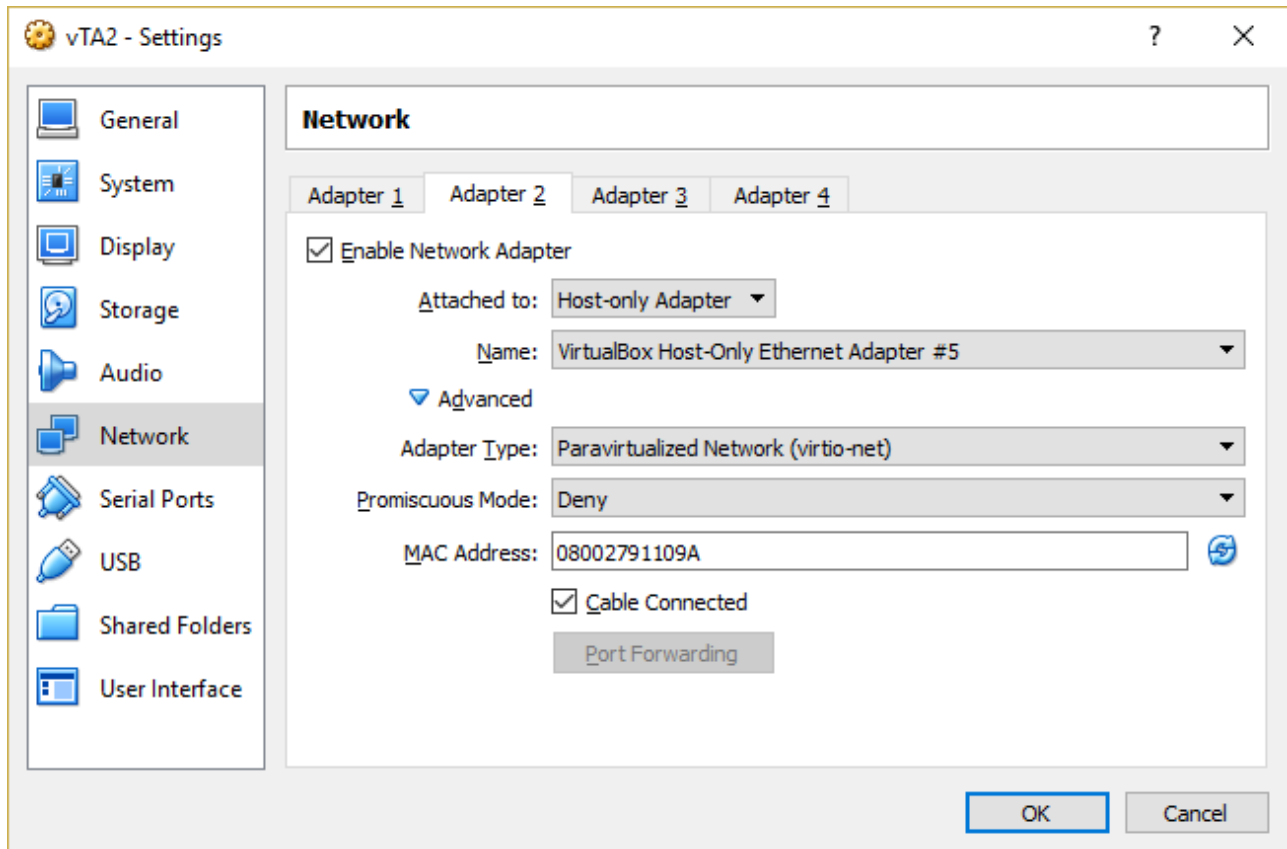


Connecting vTAs to a Host-only Ethernet Adapter

For each vTA in a subset that should be able to communicate with each other, do as follows:

- Power off the vTA if it is currently running.
- With the vTA selected in the main window, click **Settings**.
- In the left-hand pane, select **Network**.
- The **Adapter 1** tab defines the default network adapter which is attached to NAT. This network adapter can be left as-is.
- Click the **Adapter 2** tab. Here we will define a second network adapter for the vTA and attach it to the host-only adapter created in the section ["Setting Up a VirtualBox Host-only Ethernet Adapter" on page 6](#).
- Check the **Enable Network Adapter** box.
- Under **Attached to**, select **Host-only Adapter**.
- Under **Name**, select the adapter that you created in the section ["Setting Up a VirtualBox Host-only Ethernet Adapter" on page 6](#).
- Expand the **Advanced** section.
- Under **Adapter Type**, select "Paravirtualized Network (virtio-net)".
- Make sure that **Cable Connected** is checked.
- Finish by clicking **OK**.

Refer to the screenshot below.



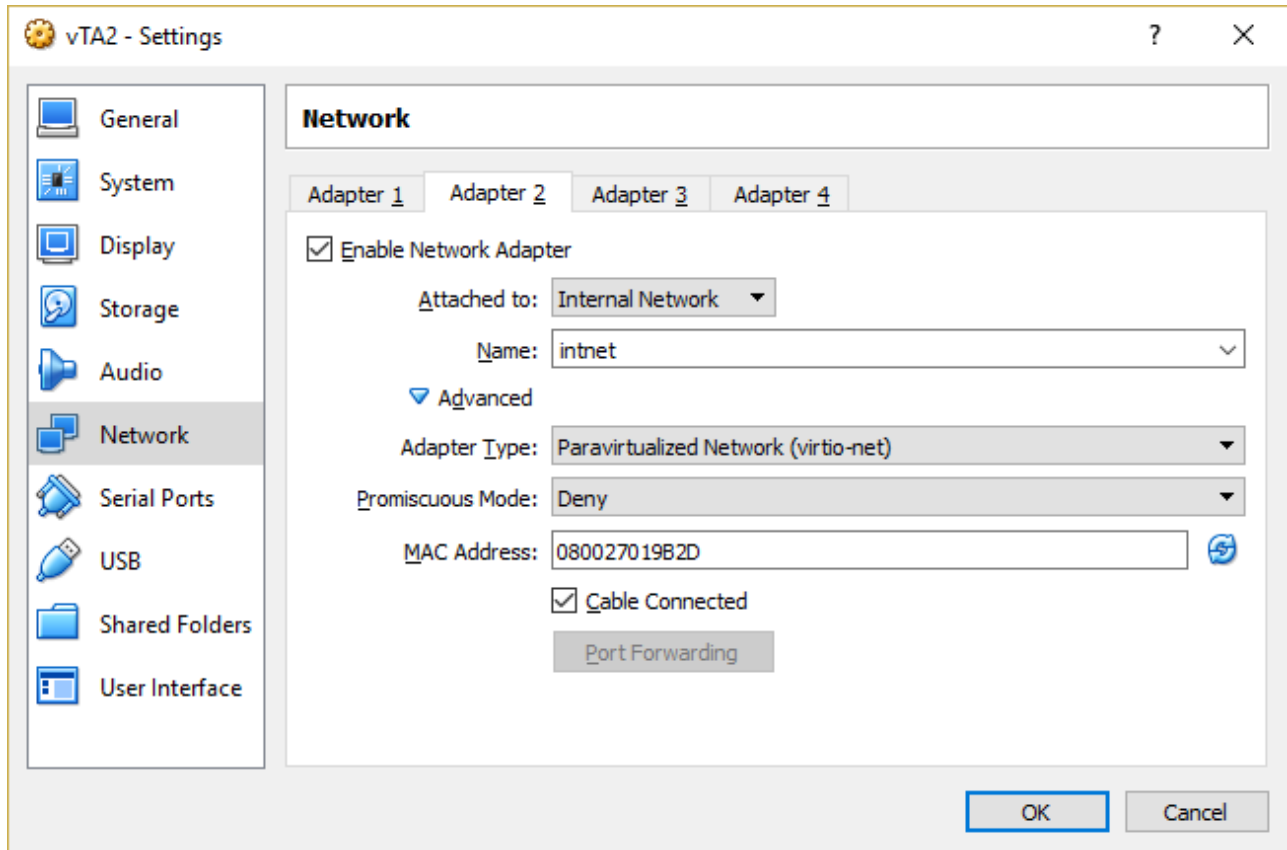
Connecting vTAs to an Internal Network Ethernet Adapter

If you are using an Internal Network adapter, here is how to set up the vTAs:

- Navigate to the vTA network adapters as described in the section ["Connecting vTAs to a Host-only Ethernet Adapter" on page 9](#).
- Check the **Enable Network Adapter** box.
- Under **Attached to**, select **Internal Network**.
- Expand the **Advanced** section.
- Under **Adapter Type**, select "Paravirtualized Network (virtio-net)".
- Make sure that **Cable Connected** is checked.

- Finish by clicking **OK**.

Refer to the screenshot below.



[1] This type of adapter allows you to access the vTA virtual machines directly from the host OS, which is convenient especially in a lab environment, where security is not a primary concern. VirtualBox also offers the option **Internal Network**, which does not permit any direct connection between the virtual machines and the host (but is otherwise similar to **Host-only Adapter**). You may wish to use the Internal Network option if you want an entirely closed network inside VirtualBox, not accessible from external hosts. See the section "[Connecting vTAs to an Internal Network Ethernet Adapter](#)" on page 10 for instructions on how to connect the vTA to an Internal Network adapter.

Appendix: Description of the vTA VNF and Its Requirements

1. The vTA VNF is capable of running in a plain, “vanilla” environment using a standard cloud configuration and orchestration based on VirtualBox. There might be some limitations in terms of

performance and also some minor limitations in terms of jitter and delay accuracy, depending on your VirtualBox infrastructure and how heavily loaded it is. However, for early proof-of-concepts and evaluations, this should not be a major issue. To obtain line rate packet generation and optimal usage of your specific hypervisor environment, an integration project would be required.

2. The vTA VNF consists of a single stand-alone VNF. However, the VNF must be able to connect and communicate securely with Paragon Active Assurance Control Center, which is not a VNF. Control Center is readily available in the public cloud (in addition to private cloud installations), something which simplifies test and evaluation projects.
3. Interfaces trust the natural OS bootstrap order in terms of how they are identified.
4. The performance is dependent on the underlying hardware. The more powerful the hardware, the higher the performance. For a 3 GHz quad-core processor, achievable performance is up to 10 Gbit/s using five concurrent unidirectional TCP streams.
5. The minimum recommended specification is: 1 vCPU, 512 MB RAM, and 2 GB of block storage. The latter two settings are in the OVF file which is part of an OVA format vTA image, as mentioned in the section ["vTA Image" on page 2](#).
6. It is assumed that a generic VNF manager which is not part of the Paragon Active Assurance solution does the instantiation, scaling, and termination of the vTA VNF.
7. The vTA VNF needs to register with Control Center to receive commands. For public cloud Control Center scenarios, the VNF needs connectivity to the Internet from the eth0 interface. For plug-and-play configuration of the VNF, DHCP should be used for IP addressing of the vTA's interfaces, as well as for assignment of an available DNS server.
8. The VNF will resolve the Control Center address and initiate an outbound connection using TCP. (For details, see the Control Center support documentation.) To successfully connect and authenticate itself to the correct Paragon Active Assurance account, the VNF needs to have credentials provided to it through the vTA virtual machine console in VirtualBox, as detailed in the chapter ["Setting Up a Virtual Test Agent in VirtualBox" on page 2](#). Once the VNF has connected to Control Center, it can be controlled either via a web browser or through the Paragon Active Assurance cloud API to start monitoring user experience KPIs, conduct a service turn-up test, or perform on-demand troubleshooting tests. The connection is an encrypted OpenVPN connection.
9. The vTA VNF also requires synchronization to an NTP server in order to achieve accurate delay and jitter measurements. By default, Test Agents will synchronize their internal clock to time.google.com, a service provided by Google; however, any NTP server (internal or external) can be used.
10. Rescaling of the VNF again needs to be handled by a generic VNF manager (compare paragraph 6). For example, if the available connection bandwidth is increased, the VNF might need to be scaled up to be able to push enough bandwidth through the link for testing purposes.