HCIbench:

Virtual SAN Automated Performance Testing Tool User Guide

Table of Contents

Introduction

Evaluating performance is an important part of considering any storage solution. Higher performing solutions can support more workloads on a given configuration, better accommodate application, minimize potential performance problems, as well as be more cost-effective. There are strong motivations to prefer higher performing solutions over lesser alternatives.

Unfortunately, obtaining directly comparable performance results from publicly available information is difficult at best. There are an infinite variety of potential test scenarios -- and many vendors discourage publishing for marketing and competitive reasons.

This leaves IT professionals in the position of having to run their own tests, and interpreting the results. This has long been a standard practice in evaluating external storage arrays, but the newer generation of hyperconverged solutions – such as VMware Virtual SAN™ -- presents new testing challenges.

In a hyperconverged architecture, each server is intended to support both many application VMs, as well as contribute to the pool of storage available to applications. This is best modeled by invoking many dozens of test VMs, each accessing multiple stored VMDKs. The goal is to simulate a very busy cluster.

Unfortunately, popular storage performance testing tools do not directly support this model. To achieve a simulation of a busy production cluster, much effort is required to automate load generation, monitoring and data collection after the fact. This takes away from the valuable time available to do actual testing, and can introduce errors into the process.

To address this situation, VMware has created a storage performance testing automation tool – HCIbench – that automates the use of the popular Vdbench testing tool in larger clusters. Users simply specify the parameters of the test they would like to run, and HCIbench instructs VDbench what to do on each and every node in the cluster.

HCIbench aims to simplify and accelerate customer Proof of Concept (POC) performance testing in a consistent and controlled way. The tool fully automates the end-to-end process of deploying test VMs, coordinating workload runs, aggregating test results, and collecting necessary data for troubleshooting purposes. Evaluators choose the profiles they are interested in; HCIbench does the rest quickly and easily.

This tool is provided free of charge and with no restrictions. Support will be provided solely on a best-efforts basis as time and resources allow, via the VMware VSAN Community Forum here: https://communities.vmware.com/community/vmtn/vsan

Per the VMware EULA, users who wish to publicly share their testing results are requested to submit their hardware configuration, methodology, parameter file and test results for review prior to publication at vsanperformance@vmware.com.

We'll make every effort to get back to you quickly.

Overview

HCIbench Tool Architecture

The tool is specifically designed for running performance tests using Vdbench against a Virtual SAN datastore. It is delivered in the form of Open Virtualization Appliance (OVA) that includes the following components:

- The test Controller VM installed with:

  - Ruby vSphere Console (RVC)

  - Virtual SAN Observer

  - Automation bundle

  - Configuration files

- Linux test VM template

The Controller VM has all the needed components installed. The core component is RVC (https://github.com/vmware/rvc) with some extended features enabled. RVC is the engine of this performance test tool, responsible for deploying Vdbench Guest VMs, conducting Vdbench runs, collecting results, and monitoring Virtual SAN by using Virtual SAN Observer.

During the installation process, you will be asked to download the Vdbench binaries directly from the Oracle website one time only. While the use of Vdbench is unrestricted, Oracle does not provide redistribution rights in their license.

The automation bundle, consisting of Ruby and Bash scripts, is developed to modularize features such as test VM deployment, VMDK initialization, and Vdbench runs, as well as automate and simplify the entire testing process. The automation bundle reads user-defined configuration information about the test environment and the target workload profile, then interacts with RVC as necessary to carry out the following tasks:

- Connect to the Virtual SAN environment to be tested. The tool itself can be deployed in a separate vSphere environment but must have access to the Virtual SAN cluster.
- Deploy Linux test VMs in the Virtual SAN cluster based on user input of the number of test VMs and the number of vDisks per VM.
- Optionally run the dd command on each vDisk to initialize storage, a similar way to "thick provisioning eager zero" or sequentially writing to storage before benchmarking.
- Transfer Vdbench parameter file to each test VM. The parameter file defines the target workload and runtime specification.
- Start Virtual SAN Observer before testing and generate Virtual SAN statistics upon test completion.
- Kick off Vdbench instances against each vDisk on each test VM and run for the defined duration.
- Collect and aggregate Vdbench performance data.

Figure 1 shows the architecture of the tool and its components.
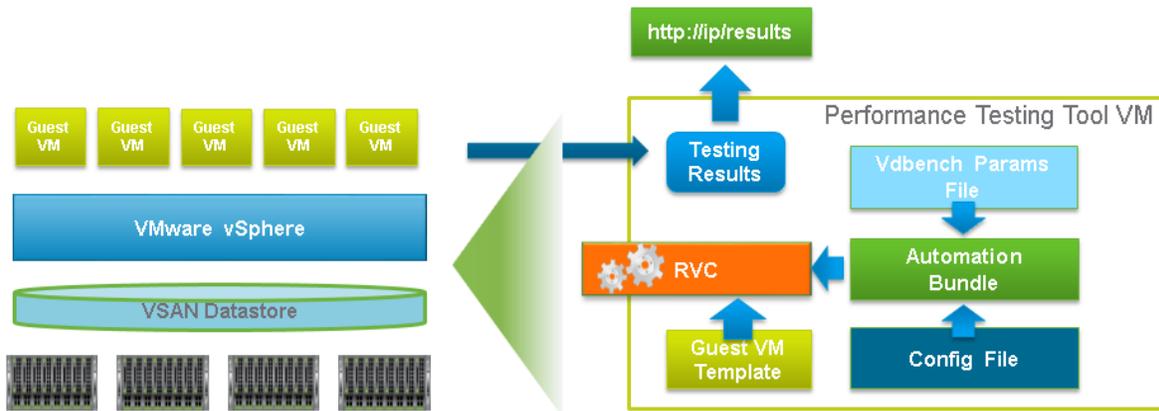
Figure 1. Tool Architecture

VM Specification

Controller VM

- CPU: 8 vCPU
- RAM: 8GB
- VMDK: 20GB
- Operating system: CentOS 6.4 64bit
- OS Credential: user is responsible for creating the root password when the VM is first powered on.
- Software installed: Ruby 1.9.3, Rubygem 2.4.5, Rbvmomi 1.8.2, RVC 1.8.0, sshpass 1.05, Apache 2.2, Tomcat 7.0

Vdbench Guest VM

- CPU: 4 vCPU
- RAM: 4GB
- OS VMDK: 2GB
- OS: Ubuntu 12.04.3 LTS 32bit
- OS Credential: root/vdbench
- Software installed: OpenJDK 1.7.0
- SCSI Controller Type: VMware Paravirtual
- Data VMDK: number and size to be defined by user

Installation and Configuration

Prerequisites

Before deploying this performance test tool packaged as OVA, make sure the environment meets the following requirements:

- The Virtual SAN Cluster is created and configured properly
- The network for Vdbench Guest VMs is ready, and must have DHCP enabled
- The vSphere environment where the tool is deployed can access the Virtual SAN Cluster environment to be tested

The tool can be deployed into any vSphere environment. However, we do not recommend deploying it into the Virtual SAN Cluster that is tested to avoid unnecessary resource consumption by the tool.

Tool Installation

Deploy the **Auto-Perf-Tool.ova** to a vSphere environment. Power on the Controller VM after deployment and set the root password when it is prompted as shown in Figure 2.

Figure 2. Set Root Password

After setting the root password, configure network for the Controller VM. Either DHCP or Static IP can be used depending on how the vSphere environment is configured. Keep the default setting and press **Ok** when "What service should be automatically started" is asked.
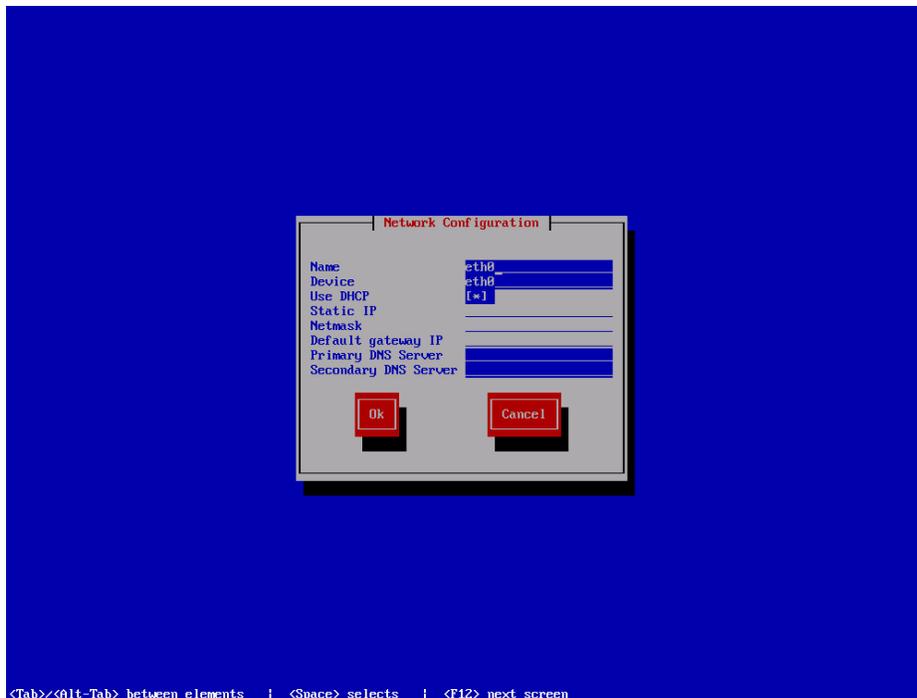
Figure 3. Controller VM Network Configuration

Tool Configuration

After deployment, you can navigate to http://Controller_VM_IP:8080/ to start configuration and kick off the test.

There are three main sections in this configuration file:

- vSphere Environment Information

    In this section, all the parameters are required except for **Network Name**. You must provide the vSphere environment information where the Virtual SAN Cluster is configured, including vCenter IP address, vCenter credential, name of the datacenter, and name of the Virtual SAN Cluster.

    - The **Network Name** parameter defines which network the Vdbench Guest VMs should use. The default value is VM Network.

    - The **Datastore Name** parameter is the datastore that is tested against and all the Vdbench Guest VMs are deployed on. Virtual SAN Datastore is used as default. If you want to conduct performance testing against another type of shared storage (NFS storage for example), enter the name of the NFS datastore.

Performance Automation Tool Configuration Page

vSphere Environment Information

vCenter Hostname/IP

| 10.143.7.108 | *

vCenter Username

| administrator@vsphere.local | *

vCenter Password

| •••••• | *

Datacenter Name

| LabDC | *

Cluster Name

| LabCluster | *

Network Name

| |

Datastore Name

| |

Figure 4. Specify vSphere Environment Information

- Virtual SAN Cluster Hosts Information

   If this parameter is **unchecked**, all of the other parameters in this section can be ignored. In this mode, a Vdbench Guest VM is deployed by the vCenter and then is cloned to all hosts in the Virtual SAN cluster in a round-robin fashion. The naming convention of Vdbench Guest VMs deployed in this mode is "vdbench-vc-<*TIME_STAMP*>-<#>".

   If this parameter is **checked**, then all the other parameters must be specified properly.

   The **Hosts** parameter specifies IP addresses or FQDNs of hosts in the Virtual SAN cluster to have Vdbench Guest VMs deployed, and all these hosts should have the same username and password specifed in **Host Username** and **Host Password**. In this mode, Vdbench Guest VMs are deployed directly on the specified hosts concurrently. To reduce the network traffic, five hosts are running deployment at the same time then it moves to the next five hosts. Each host also deploys at an increment of five VMs at a time. The naming convention of test VMs deployed in this mode is "vdbench-<*TIME_STAMP*>-<*HOSTNAME/IP*>-batch<*VM#*>-<*VM#*>".

   In general, for deployment of a large number of test VMs, it is recommneded to **check Deploy on Hosts.**

   Note: If Distributed vSwitch PortGroup is used for Network, **Deploy on Hosts** must be **unchecked**.

VSAN Cluster Hosts Information

Directly Deploy on Hosts
☑ Deploy on Hosts

Hosts

host1.test.com
host3.test.com
host3.test.com

Host Username

root

Host Password

••••••

Figure 5. Specify Virtual SAN Cluster Nodes

- Vdbench Guest VM Specification

    In this section, the only required parameter is **Number of VMs** that specifies the total number of Vdbench Guest VMs to be deployed for testing. The other two parameters **Number of Data Disk** and **Size of Data Disk** are optional:

    - The **Number of Data Disk** parameter specifies how many VMDKs to be tested are added to each Vdbench Guest VM.

    - The **Size of Data Disk** parameter specifies the size(GB) of each VMDK to be tested. The total number of simulated workload instances is **Number of VM** times **Number of Data Disk**.

    The default value of both parameters is 10. You should take a carefule sizing exercise to make sure there is sufficient compute and storage resoruces to support the target level of workload instances.

Vdbench Guest VM Specification

Number of VMs

2   *

Number of Data Disk

10

Size of Data Disk

10

Figure 6. Specify Test VM Information

Vdbench Workload Configuration

You can get the Vdbench parameter file in the following ways:

Figure 7. Vdbench Workload Configuration

- Select a Vdbench file in the drop-down box.

  If a parameter file is uploaded or generated to the controller before, it already exists in the controller. In this case, you can select the existing Vdbench parametner file and reuse it. You can also refresh the drop-down list by clicking the **Refresh** button. After you finish generating a parameter file or uploading a parameter file, click the **Refresh** button and it makes the file displayed in the drop-down list without refreshing the entire page to avoid user-input lost. Delete the parameter file by clicking the **Delete** button.

- Upload a Vdbench parameter file

  If the desired parameter file does not exist, you can create a self-defined parameter file and upload it to the controller by clicking the **Choose File** button in the **Upload a Vdbench Parametner File** section. For Vdbench parameter file format, refer to the Vdbench Users Guide.

- Generate a Vdbench parameter file

  If you do not want to edit a Vdbench parameter file by hand, which is error prone, this tool also provides a **Vdbench Parameter File Generate Page** for you to easily create a self-defined parameter file through the GUI. By clicking the **Generate** button, you are redirected to the following page:

Figure 8. Vdbench Parameter File Generate Page

Note: The value of **Number of Data Disk** in the Vdbench Guest VM Specification section must match the value of **Number of Disks to Test** defined in the Vdbench parameter files. For example, if you specify to create 10 data disks per Guest VM, 10 raw disks /dev/sdb - /dev/sdk is created. Therefore, in the Vdbench parameter files, the same number or less of disks /dev/sdb - /dev/sdk are expected.

Refer to the Vdbench Users Guide for configuration of workload parameter files.

Figure 9. Other Parameters

The **Output Path** parameter is another local directory on the Controller VM for storing collected results from all Vdbench Guest VMs and statistics produced by Virtual SAN Observer. If not specified, the default path **"/opt/output/results**" is applied. It is recommended to use the default path since this directory could be browsed at http://Controller_VM_IP/results in a Web browser.

Users can choose whether to intialize the data VMDKs of test VMs by checking the **Initialize Storage** parameter.

The **Testing Duration** parameter is for overriding the elapsed value in Vdbench parameter files. This parameter defines the test duration for each run. If not specified, each Vdbench test run uses its own elapsed value.

When **Clean up VMs** parameter is **checked**, all the client VMs are removed after all the testing is completed; otherwise, all the VMs are perserved.

Download Vdbench

Before running the test, you must download the Vdbench tool from the Oracle website.

Click the **Download** button. After the download is completed, you should upload the zipped file. And the server unzips the Vdbench file to /opt/vm-template. This step is a once-for-all action. The following screen disappears from the page after you upload the Vdbench file successfully.

Download the Vdbench

[Download]

Download the Vdbench from the Oracle website. And put the file into the directory

Upload the Vdbench File

[Choose File] no file selected

Figure 10. Download the Vdbench

Save Configuration

Press the **Save Configuration** button to save the parameter configuration settings. If the configuration setting is not saved, the system uses the previous saved parameter configuration by default.

Configuration Validation

After completing the tool configuration, you can validate all settings by clicking the **Validate** button. This step checks if all the required information is correctly provided. Additionally, it validates basic environment sanity including whether Virtual SAN is enabled in the cluster, whether the hosts specified belong to the cluster and can access the Virtual SAN datastore. Furthermore, this function estimates the storage usage by all Guest VMs on the Virtual SAN datastore and alert if it exceeds 80 percent of the usable capacity after deployment.

2015-07-08 00:23:41 -0700: Validating VC IP and Crendetial...
2015-07-08 00:23:43 -0700: VC IP and Credential Validated
2015-07-08 00:23:43 -0700: Validating Datacenter Datacenter...
2015-07-08 00:23:44 -0700: Datacenter Datacenter Validated
2015-07-08 00:23:44 -0700: Validating Cluster Cluster1...
2015-07-08 00:23:46 -0700: Cluster Cluster1 Validated
2015-07-08 00:23:46 -0700: Validating Network VM Network...
2015-07-08 00:23:48 -0700: Network VM Network Validated
2015-07-08 00:23:48 -0700: Validating If VSAN Enabled in Cluster Cluster1...
2015-07-08 00:23:54 -0700: VSAN is Enabled in Cluster Cluster1, the VSAN Datastore name is vsanDatastore, capacity is 33180 GB and freespace is 27748 GB
2015-07-08 00:23:56 -0700: Deploy on hosts: False. Skip validating hosts...

2015-07-08 00:23:56 -0700: All the config has been validated, please go ahead to kick off testing
----

Figure 11. Configuration Validation

2015-07-08 00:26:32 -0700: Validating VC IP and Crendetial...

VC 10.110.186.32 IP or Credential Info incorrect!

Figure 12. Validation Failure

After the validation is successfully completed, a message is displayed to inform you that you can continue with the testing.

Tool Usage

How to Run Tests

You can click the **Test** button to start the program. The testing is a time-consuming operation with the test progress displayed on the web page.
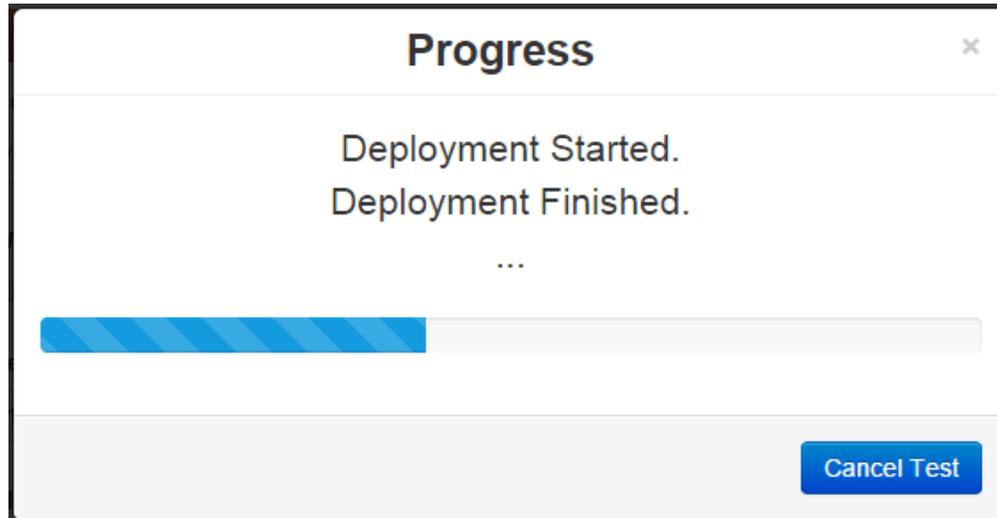
Figure 13. Test in Progress

During the testing, you can kill the test process by clicking the **Cancel Test** tab**.**

How to Consume Test Results

After the Vdbench testing is completed, the test results are collected from all Vdbench instances in the test VMs. And you can find the results in the **Output Path** directory specified in the configuration file. If the default path is used, the results could be viewed at http://Controller_VM_IP/results in a web browser.

Index of /results

Name	Last modified	Size	Description
Parent Directory		-	
results20150706034336/	06-Jul-2015 22:13	-	
results20150706232244/	06-Jul-2015 23:22	-	
results20150706232314/	06-Jul-2015 23:23	-	
results20150706232336/	06-Jul-2015 23:46	-	
results20150707002445/	07-Jul-2015 00:24	-	
results20150707012349/	07-Jul-2015 01:23	-	
results20150707014529/	07-Jul-2015 01:45	-	
results20150707021231/	07-Jul-2015 02:12	-	
results20150707024331/	07-Jul-2015 02:43	-	
results20150707135010/	07-Jul-2015 13:50	-	
results20150707135033/	07-Jul-2015 13:50	-	
results20150707135201/	07-Jul-2015 13:52	-	
results20150707135913/	07-Jul-2015 13:59	-	
results20150707135926/	07-Jul-2015 13:59	-	
results20150707194449/	07-Jul-2015 19:44	-	
results20150707194616/	07-Jul-2015 19:46	-	
results20150707194722/	07-Jul-2015 19:47	-	
results20150708001215/	08-Jul-2015 00:12	-	
results20150708002240/	08-Jul-2015 00:22	-	
results20150708002536/	08-Jul-2015 00:25	-	
results20150708002903/	08-Jul-2015 00:29	-	

Apache/2.2.15 (CentOS) Server at 10.110.186.119 Port 80

Figure 14. Test Results

Each of the sub-directories in the **Output Path** directory uses the name of the user-defined Vdbench parameter file, and contains all original results produced by each Vdbench instance and Virtual SAN Observer data.

The aggregate result of one test run is summarized in the text file with the name <DIR_NAME>-res.txt, containing four statistics: number of VMs used for testing, IOPS, throughput, and latency.

```
VMs   =        7
IOPS  =    86682.54 IO/s
TPUT  =      507.68 MB/s
LAT   =        1.00 ms
```

Figure 15. Aggregated Performance Data

All of the original result files produced by Vdbench instances can be found inside the sub-directory corresponding to a test run. In addition to the text files, there is another sub-directory named iotest

vdbench-<*VM#*>vm inside, which is the statistics directory generated by Virtual SAN Observer.

Figure 16. Detailed Performance Data

Open the stats.html file inside the statistics directory, you can find the Virtual SAN performance statistics for debugging or evaluating purposes.
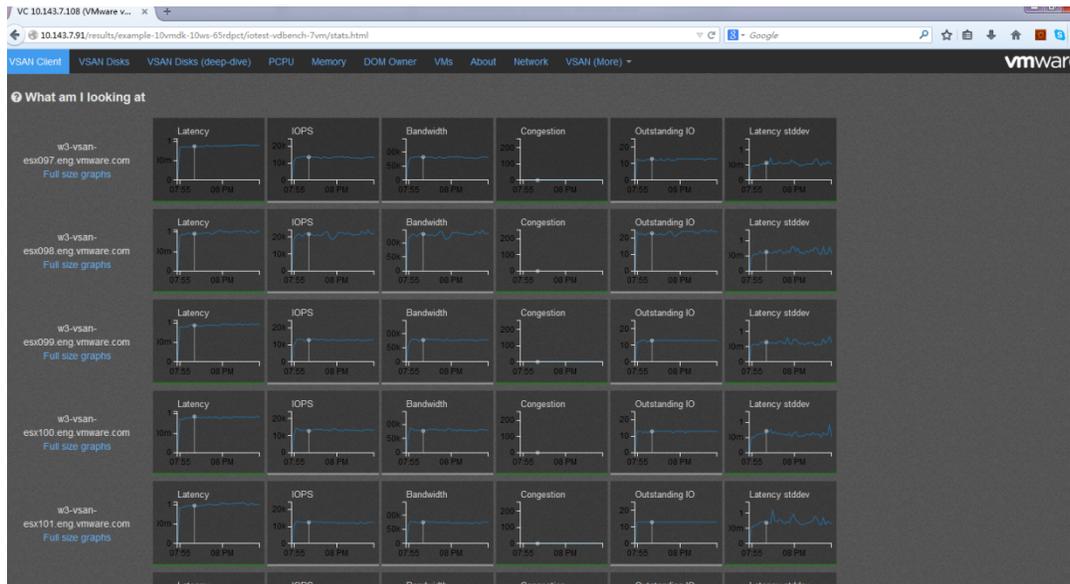
Figure 17. Virtual SAN Observer Statistics

How to Download Test Results to Local Disk

Download the test results by clicking the **Save Result** button. The latest test result details are zipped to a file and you can download the file to your local client.

About the Author and Contributors

Chen Wei, Victor Chen, and Dave Yu in the VMware Product Enablement team wrote the original version of this paper. Catherine Xu, technical writer in the Product Enablement team, edited this paper to ensure that the contents conform to the VMware writing style.

vmware®