HCIBench User Guide

vmware

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 to lesser alternatives.

Unfortunately, obtaining directly comparable performance results from publicly available information is difficult at best. There is 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 hyper-converged solutions – such as VMware vSAN™—presents new testing challenges.

In a hyper-converged 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 kind of 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. These steps waste so much valuable time available to do actual testing, even worse may introduce errors into the process.

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

HCIBench aims to simplify and accelerate customer Proof of Concept (POC) performance testing in a consistent and controlled way. This 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-effort basis as time and resources allow, by the VMware vSAN Community Forum.

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

We will 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 vSAN 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)
  - vSAN 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 vSAN by using vSAN Observer.

During the installation process, you need 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 vSAN environment to be tested. The tool itself can be deployed in a separate VMware vSphere® environment but must have access to the vSAN Cluster.
- Deploy Linux test VMs in the vSAN Cluster based on user input of the number of test VMs and the number of virtual disks per VM.
- Optionally run the fio tool on each virtual disk 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 vSAN Observer before testing and generate vSAN statistics upon test completion.
- Kick off Vdbench instances against each virtual disk on each test VM and run for the defined duration.
- Collect and aggregate Vdbench performance data.
- Execute vSAN Performance Diagnostics if running against vSAN 6.6.1 with CEIP (Customer Experience Improvement Program) turned on.

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

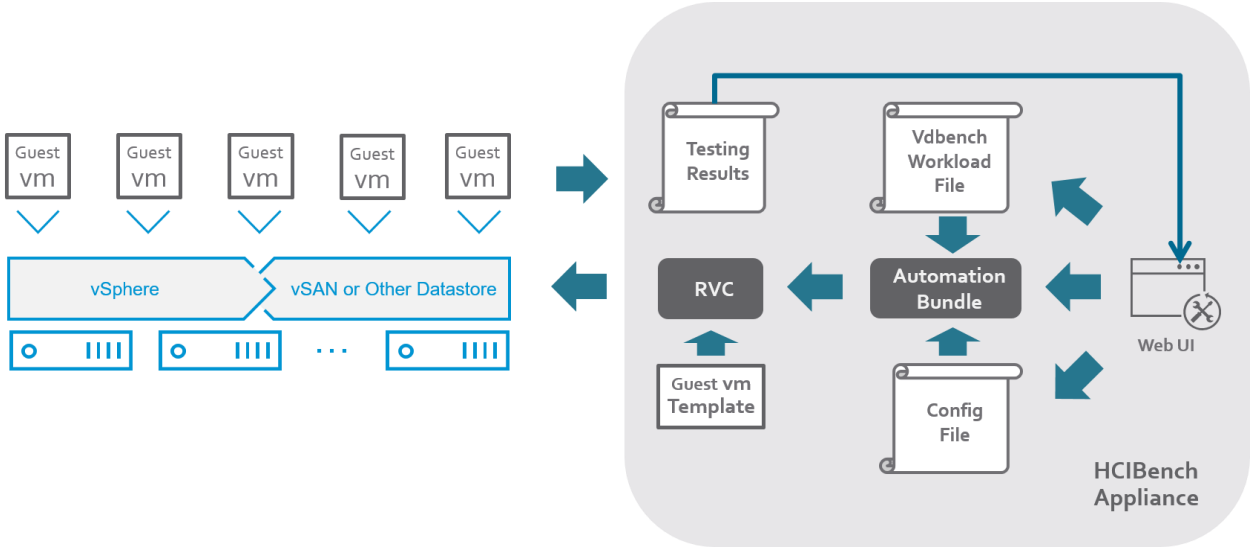

Figure 1. Tool Architecture

VM Specification

Controller VM

- CPU: 8 vCPU
- RAM: 8GB
- OS VMDK: 16GB
- Operating system: Photon OS 1.0
- OS Credential: user is responsible for creating the root password when deploying the VM.
- Software installed: Ruby 2.3.0, Rubygem 2.5.1, Rbvmomi 1.11.3, RVC 1.8.0, sshpass 1.05, Apache 2.4.18, Tomcat 8.54, JDK 1.8u102

Vdbench Guest VM

- CPU: 4 vCPU
- RAM: 8GB
- OS VMDK: 16GB
- OS: Photon OS 1.0
- OS Credential: root/vdbench
- Software installed: JDK 1.8u102, fio 2.13
- 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 vSAN Cluster is created and configured properly.

- The network for Vdbench Guest VMs is ready, and needs to have DHCP service enabled; if the network doesn't have DHCP service, "Private Network" must be mapped to the same network when HCIBench being deployed.
- The vSphere environment where the tool is deployed can access the vSAN Cluster environment to be tested.

The tool can be deployed into any vSphere environment. However, we recommend deploying it into the vSAN Cluster that is tested to avoid potential networking issue.

Tool Installation

Deploy the **HCIBench.ova** to a vSphere environment, accept the EULA of Photon OS and choose thin provision for disk format, then map the "Public Network" to the network which the HCIBench will be accessed through; if the network prepared for Vdbench Guest VM doesn't have DHCP service, map the "Private Network" to the same network, otherwise just ignore the "Private Network".

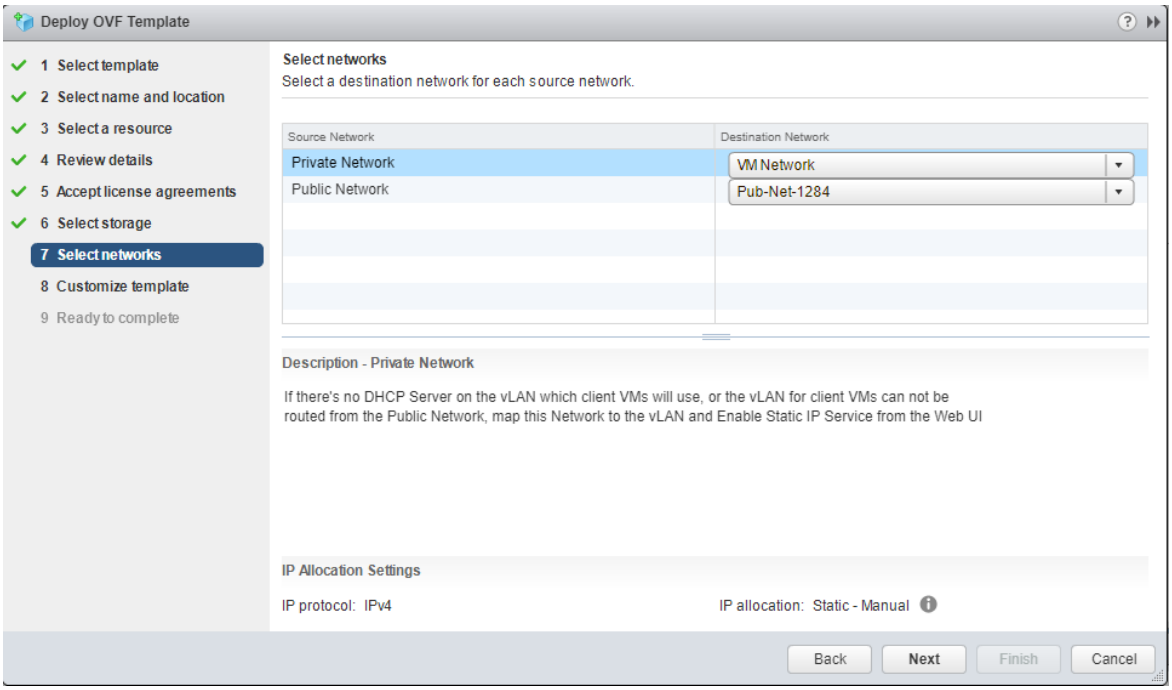

Figure 2. Map Networks

After mapping the networks configure the public network and the system password of HCIBench in the next page. The network information could be left blank if DHCP is selected.

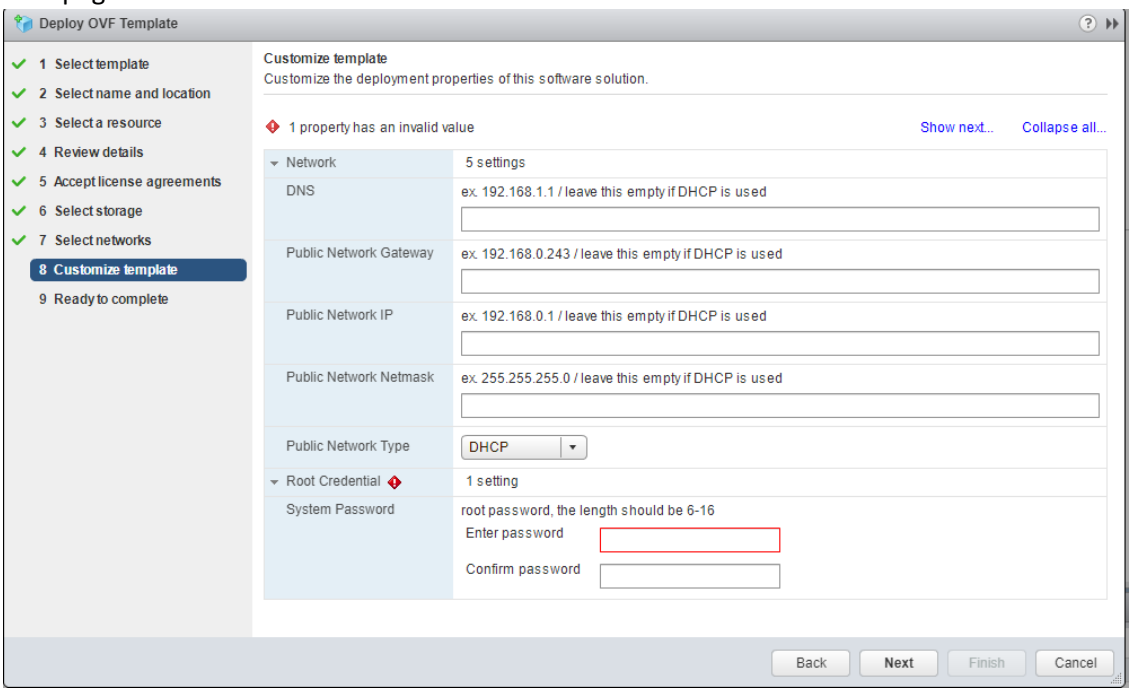

Figure 3. Configure Public Network and System Password

Tool Configuration

After deployment, you can navigate to http://HCIBench_IP:8080/ to start configuration and kick off the test. Before accessing the configuration page, the root user ID and password must be used to authenticate to prevent unauthorized access to HCIBench.

There are three main sections in this configuration file:

- vSphere Environment Information

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

  - The **vCenter Username** should use the vSphere account which has the adminitrator permission of the vSphere environment.
  - The **Network Name** parameter defines which network the Vdbench Guest VMs should use. The default value is **VM Network**.
  - If **Enable DHCP Service on the Network** parameter is checked, HCIBench will start DHCP service on the VLAN which the "Private Network" mapped on.
    NOTE: From HCIBench 1.6.5, we stop providing DHCP service from HCIBench and use Static IP instead. By checking **Set Static IP for Vdbench Client VMs**, HCIBench will pick up one available IP address from 192.168.2.1 to 192.168.2.254 for the private network interface (eth1) and set the client VMs using from 192.168.3.0 to 192.168.63.255. Every single client VM does arping before

setting up the IP address for itself to avoid IP conflict. In this case, you can have multiple HCIBench instances using the same internal vLAN to deploy client VMs to conduct performance testing against different clusters at the same time.

- The **Datastore Name** parameter specifies the datastores that are tested against and all the Vdbench Guest VMs are deployed on. You need to enter the name of the datastore. Testing multiple datastores in parallel is also supported. You can enter the datastore names one per line. In this cases, the virtual machines are deployed evenly on the datastores. For example, if you enter two datastores and 100 virtual machines, 50 virtual machines are deployed on each datastore.
- **Clear Read/Write Cache Before Each Testing** is the option desgined for vSAN user to flush the cache tier before each Vdbench test case, ESXi **Host Username** and **Host Password** must be unique and specified if this box is checked.

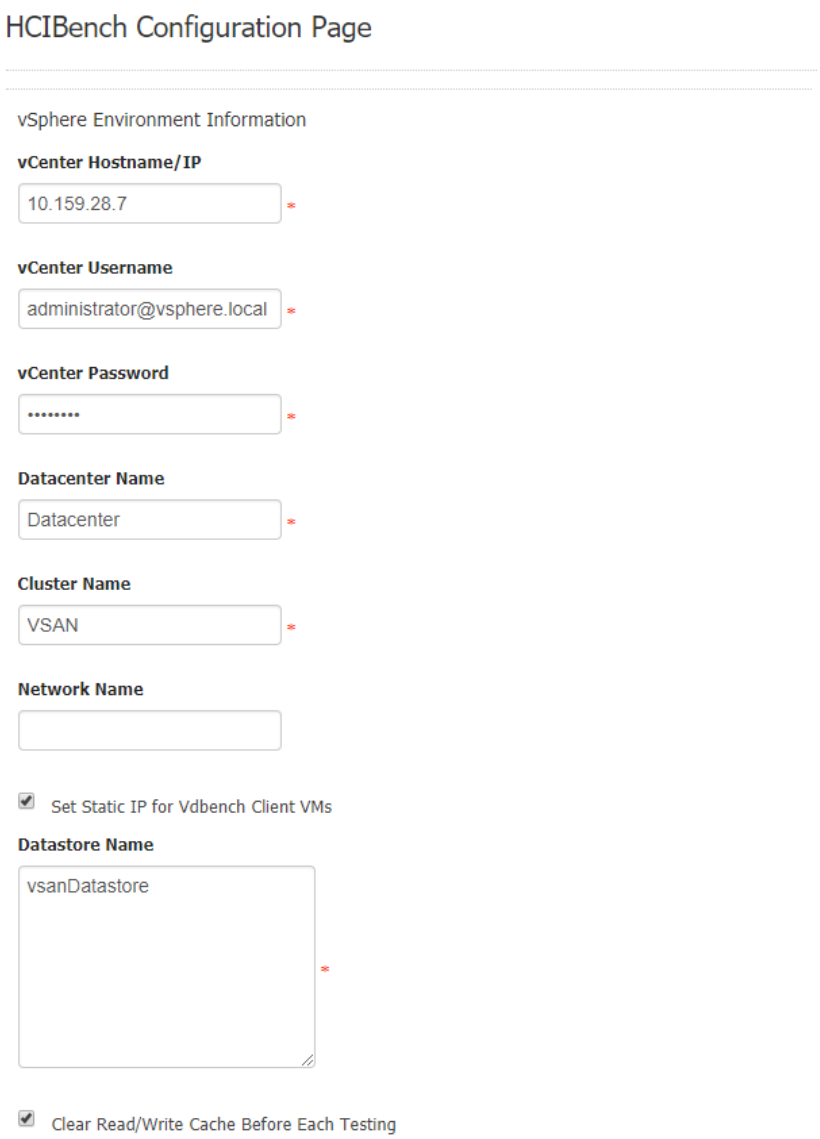

Figure 4. Specify vSphere Environment Information

- Cluster Hosts Information

  If this parameter is unchecked, ignore the **Hosts** field below, for the **Host Username/Password** fields can also be ignored if **Clear Read/Write Cache Before Each Testing** is unchecked. In this mode, a Vdbench Guest VM is deployed by the vCenter and then is cloned to all hosts in the vSAN Cluster in a round-robin fashion. The naming convention of Vdbench Guest VMs deployed in this mode is "vdbench-vc-<*DATASTORE_NAME*>-<*#*>".

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

  The **Hosts** parameter specifies IP addresses or FQDNs of hosts in the vSAN 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-<*HOSTNAME/IP*>-<*DATASTORE_NAME*>-batch<*VM#*>-<*VM#*>".

  In general, it is recommneded to check **Deploy on Hosts** for deployment of a large number of test VMs. However, if distributed switch portgroup is used as the client VM network, **Deploy on Hosts** must be unchecked.

  EASY RUN is specifically designed for vSAN user, by checking this, HCIBench is able to handle all the configurations below by identifying the vSAN configuration. **EASY RUN** helps to decide how many client VMs should be deployed, the number and size of VMDKs of each VM, the way of preparing virtual disks before testing etc. the configurations below will be hidden if this option is checked.

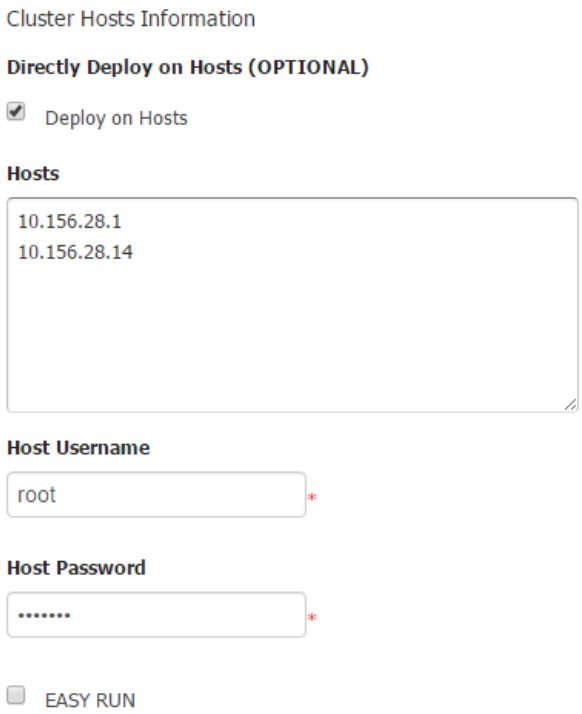

Figure 5. Specify vSAN Cluster Nodes

- 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. If you enter multiple datastores, these VMs are deployed evenly on the datastores. The **Number of Data Disk** and **Size of Data Disk** parameters 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.

Re-Use The Existing VMs If Possible allows user to reuse the Vdbench client VMs in the cluster if they are existing and compatible with the VM specification.

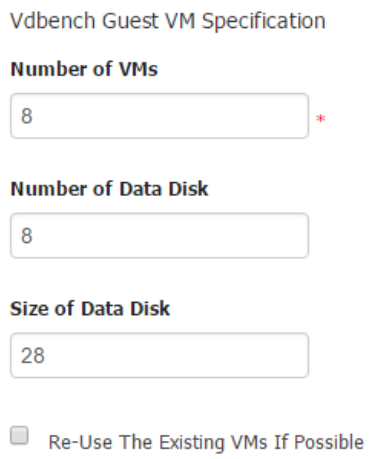

Figure 6. Specify Test VM Information

Vdbench Testing Configuration

The **Test Name** parameter is the name of the test, by specifying this parameter, for example "DemoTest", HCIBench will create a local directory with the same name under "/opt/output/results/" on the Controller VM for storing collected results from all Vdbench Guest VMs and statistics produced by vSAN Observer. If not specified, a name "resultsTIMESTAMP" will be generated and the same name directory will be created under "/opt/output/results".All the test cases results could be browsed at http://Controller_VM_IP/results.You can get the Vdbench parameter file in the following ways:

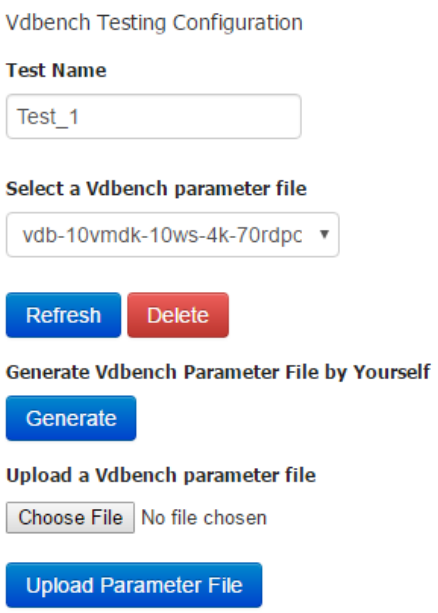

Figure 7. Vdbench Workload Configuration

- 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 loss. 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. After uploading, click **Refresh** button above, the file you uploaded will be in the drop-down list. 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 manually, 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:

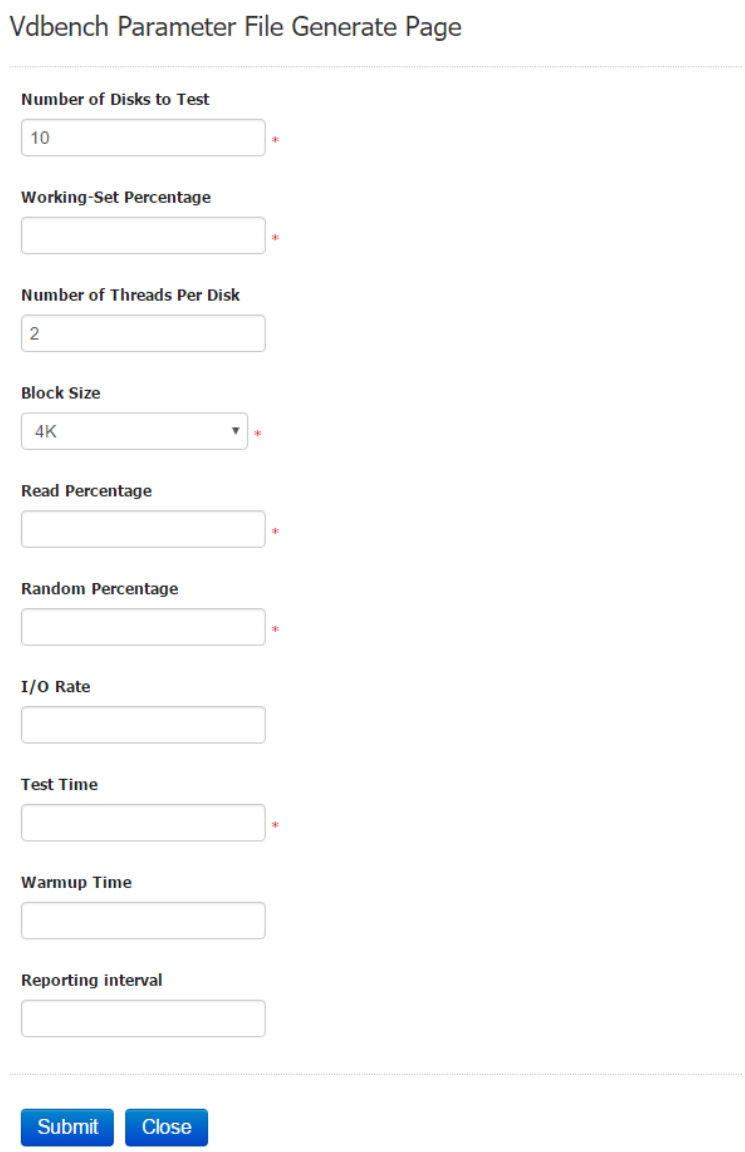

Figure 8. Generate Page of Vdbench Parameter File

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 are created. Therefore, in the Vdbench parameter files, the same number or less of disks are expected.

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

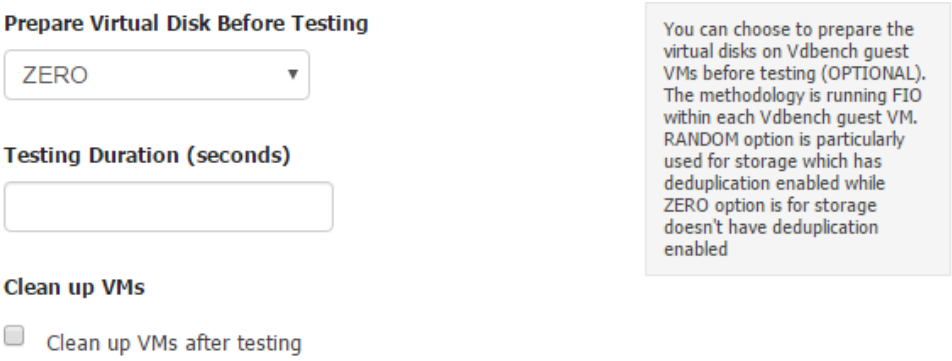

Figure 9. Other Parameters

Users can choose whether to intialize the data VMDKs of test VMs. There are two options of storage initialization, **ZERO** and **RANDOM**. **RANDOM** is particularly for storage that has de-duplication enabled, if the storage will be tested against doesn't have de-duplication enabled, use **ZERO** instead to initialize storage to avoid first-write penalty.

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 the **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 zip file. And the server will automatically put the Vdbench file to /opt/output/vdbench-source. This step is a once-for-all action. The following screen disappears from the page after you upload the Vdbench file successfully.

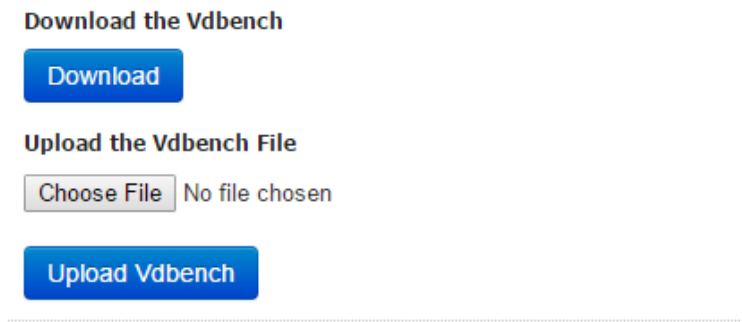

Figure 10. Download the Vdbench

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 vSAN is enabled in the cluster, whether the hosts specified belong to the cluster and can access the vSAN datastore. Furthermore, this function estimates the storage usage by all Guest VMs on the vSAN datastore and alert if it exceeds 80 percent of the usable capacity after deployment.

Information

```
2015-12-15 14:42:47 -0800: Validating VC IP and Crendetial...
2015-12-15 14:42:51 -0800: VC IP and Credential Validated
2015-12-15 14:42:51 -0800: Validating Datacenter Lab...
2015-12-15 14:42:54 -0800: Datacenter Lab Validated
2015-12-15 14:42:54 -0800: Validating Cluster VSAN...
2015-12-15 14:42:55 -0800: Cluster VSAN Validated
2015-12-15 14:42:55 -0800: Validating Network VM Network-1284...
2015-12-15 14:42:58 -0800: Network VM Network-1284 Validated
2015-12-15 14:42:58 -0800: Checking If Network VM Network-1284 is accessible from all the hosts of VSAN...
2015-12-15 14:43:05 -0800: Network VM Network-1284 is accessible from host 10.156.28.21
2015-12-15 14:43:07 -0800: Network VM Network-1284 is accessible from host 10.156.28.22
2015-12-15 14:43:10 -0800: Network VM Network-1284 is accessible from host 10.156.28.23
2015-12-15 14:43:12 -0800: Network VM Network-1284 is accessible from host 10.156.28.24
2015-12-15 14:43:12 -0800: Network VM Network-1284 is accessible from all the hosts of VSAN
2015-12-15 14:43:12 -0800: Validating Type of Network VM Network-1284...
2015-12-15 14:43:13 -0800: Network VM Network-1284 Type is Network
2015-12-15 14:43:13 -0800: Validating Datastore vsanDatastore...
2015-12-15 14:43:17 -0800: Datastore vsanDatastore Validated
2015-12-15 14:43:19 -0800: Checking Datastore vsanDatastore type...
2015-12-15 14:43:22 -0800: Datastore vsanDatastore type is vsan
2015-12-15 14:43:22 -0800: Checking If Datastore vsanDatastore is accessible from all the hosts of VSAN...
2015-12-15 14:43:27 -0800: Datastore vsanDatastore is accessible from host 10.156.28.21
2015-12-15 14:43:27 -0800: Datastore vsanDatastore is accessible from host 10.156.28.22
2015-12-15 14:43:27 -0800: Datastore vsanDatastore is accessible from host 10.156.28.23
2015-12-15 14:43:27 -0800: Datastore vsanDatastore is accessible from host 10.156.28.24
2015-12-15 14:43:27 -0800: Datastore vsanDatastore is accessible from all the hosts of VSAN
2015-12-15 14:43:27 -0800: Validating If VSAN Enabled in Cluster VSAN...
2015-12-15 14:43:35 -0800: VSAN is Enabled in Cluster VSAN, the VSAN Datastore name is vsanDatastore, capacity is 22132 GB and freespace is
13434 GB
2015-12-15 14:43:45 -0800: Deploy on hosts: False. Skip validating hosts...
--------------------------------------------------------------------------------
2015-12-15 14:43:45 -0800: All the config has been validated, please go ahead to kick off testing
--------------------------------------------------------------------------------
```

Figure 11. Configuration Validation

Information

```
2015-12-15 14:47:27 -0800: Validating VC IP and Crendetial...
--------------------------------------------------------------------------------
VC 10.156.169.96 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 toolbar displayed on the web page.

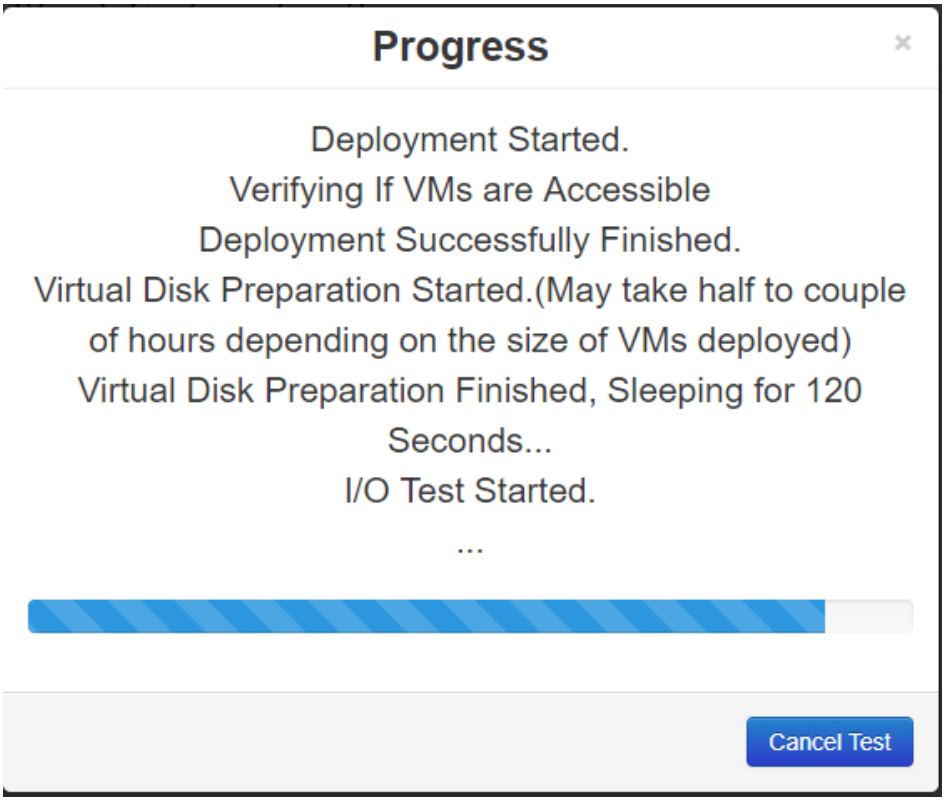

Figure 13. Test in Progress

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

How to Consume Test Results

After the Vdbench testing finishes, the test results are collected from all Vdbench instances in the test VMs. And you can view the results at http://HCIBench_IP/results in a web browser.

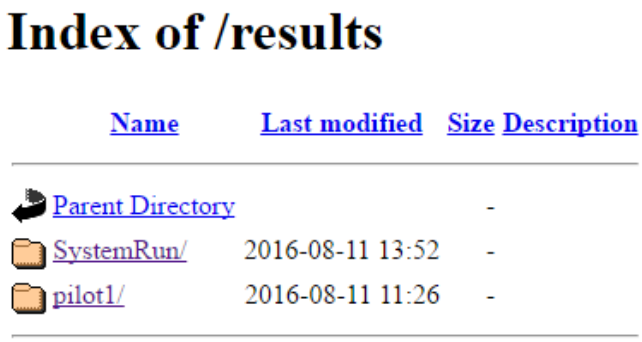

Figure 14. Test Results

Each of the subdirectories in "/opt/output/results/TestName" directory uses the name of the user-defined Vdbench parameter file, and contains all original results produced by each Vdbench instance and vSAN Observer data.

The aggregate result of one test run is summarized in the text file with the name <*DIR_NAME*>-res.txt, containing the datastore's name and four statistics: number of VMs used for testing, IOPS, throughput, latency details and host resource consumption. If vSAN is enable, the CPU used by vSAN will also be calculated.

```
Datastore:  vsanDatastore
============================
Run Def:    RD=run1; I/O rate: Uncontrolled MAX; elapsed=3600 warmup=1800; For loops: None
VMs         = 48
IOPS        = 1036544.95 IO/s
THROUGHPUT  = 4049.00 MB/s
LATENCY     = 3.0576 ms
R_LATENCY   = 2.9271 ms
W_LATENCY   = 3.3625 ms
95%tile_LAT = 4.9941 ms
============================
Resource Usage:
CPU USAGE  = 68.19%
RAM USAGE  = 15.61%
VSAN PCPU USAGE = 23.2684%
============================
If you are interested in improving the IOPS/THROUGHPUT/LATENCY, please find the details in file
performance_diag_result.html in directory vdb-8vmdk-100ws-4k-70rdpct-100randompct-8threads-1510473945
```

Figure 15. Aggregated Performance Data

You can find all the original result files produced by Vdbench instances inside the subdirectory corresponding to a test run.

If you are running testing against vSAN 6.6.1 or later version with CEIP turned on, there will be a file called **performance_diag_result.html** generated in this subdirectory. In this file, by choosing one of the three goals (**To Get More IOPS**; **To Get More Throughput**; **To Get Lower Latency**), you can see the potential issue found by VMware Cloud Server. And you can find the associated KB article by clicking **Ask VMware** and go to the vCenter vSAN cluster monitor page to find the detailed information by clicking **vCenter**.

Select the category you want to improve

To Get Lower Latency

Potential Issue: vSAN is experiencing congestion in one or more disk group(s)

Description: This implies that one or more disk groups are experiencing congestion. The disk group(s) which are experiencing congestion are listed, along with the type of congestion. Note that regardless of the type of congestion, temporary values of congestion are usually ok, and not detrimental to the clusters' performance. However, consistent values of congestion warrant attention and resolution. Consult Ask VMware for a recommendation on possible solutions. Ask VMware

Potential Issue: The outstanding IOs for the benchmark might not be optimal to achieve the desired goal

Description: This implies that the outstanding IOs are either too low or too high for the desired goal. The outstanding IOs must be high (ideally, at least 2 per host) for maximum IOs per second and maximum throughput, otherwise the system is bottlenecked by the time it takes to process IOs. On the other hand, outstanding IOs must be low (ideally 1 per host) for minimum latency, or the IOs may queue at different layers leading to higher latency. The remedy is to adjust the outstanding IOs according to the desired goal of performance evaluation. Consult Ask VMware for a recommendation on how to set the correct outstanding IO. Ask VMware

Please go to vCenter to locate the time range named HCIBench-vdb-8vmdk-100ws-64k-0rdpct-0randompct-1threads-1510006342 for more details

Figure 16. vSAN Performance Diagnostic

In addition to those files, there is another subdirectory named iotest-vdbench-<*VM#*>vm inside, which is the statistics directory generated by vSAN Observer.

Index of /results/48vms-all-80G/vdb-8vmdk-100ws-4k-70rdpct-100randompct-8threads-1510473945

Name	Last modified	Size	Description
Parent Directory		-	
iotest-vdbench-48vm/	2017-11-12 02:25	-	
performance_diag_result.html	2017-11-12 02:17	856	
vdbench-vsanDatastore-0-1.txt	2017-11-12 02:16	683K	
vdbench-vsanDatastore-0-2.txt	2017-11-12 02:16	683K	
vdbench-vsanDatastore-0-3.txt	2017-11-12 02:16	683K	
vdbench-vsanDatastore-0-4.txt	2017-11-12 02:16	683K	
vdbench-vsanDatastore-0-5.txt	2017-11-12 02:16	683K	
vdbench-vsanDatastore-0-6.txt	2017-11-12 02:16	683K	
vdbench-vsanDatastore-0-7.txt	2017-11-12 02:16	683K	
vdbench-vsanDatastore-0-8.txt	2017-11-12 02:16	683K	

Figure 17. Detailed Performance Data

Open the stats.html file inside the statistics directory, you can find the vSAN performance statistics for debugging or evaluating purposes.

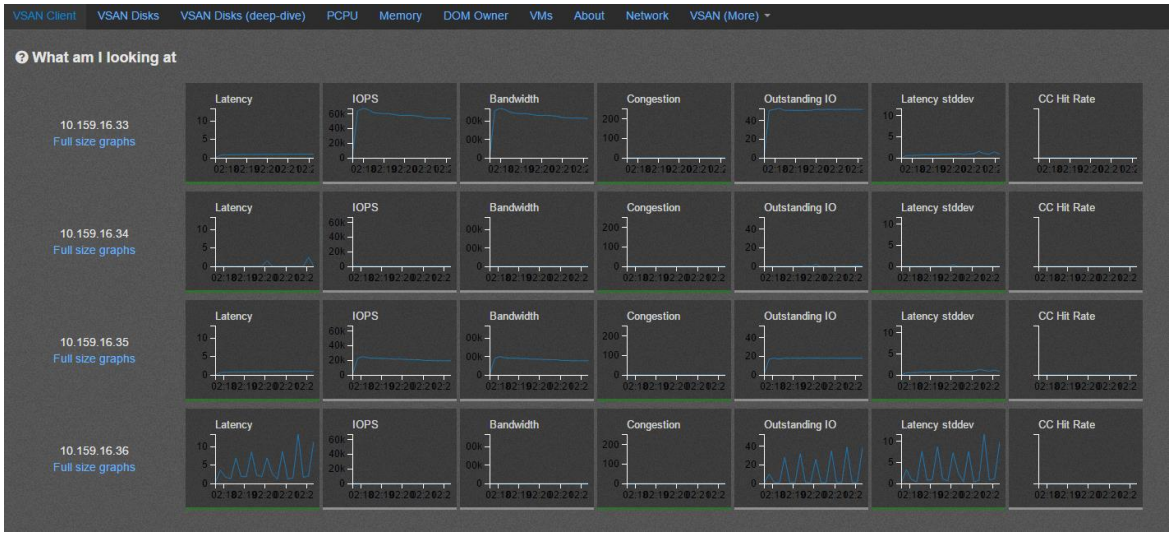

Figure 18. vSAN Observer Statistics

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

VMware, Inc. 3401 Hillview Avenue Palo Alto CA 94304 USA Tel 877-486-9273 Fax 650-427-5001 www.vmware.com