Elastic Cloud Server

Instance Performance Evaluation Guide

Issue 01

Date 2020-12-17

HUAWEI TECHNOLOGIES CO., LTD.

Trademarks and Permissions


and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Contents

1 Methods of Evaluating Instance Compute Performance.....1

2 Compute Performance Appraisals for Mainstream Instances 5

3 Change History.....8

1 Methods of Evaluating Instance Compute Performance

Scenarios

This section describes how to use the SPEC CPU®2017 benchmark suite to evaluate the compute performance of a Linux instance.

Prerequisites

- An ECS that is created using a public image on the cloud platform is available.
- At least 2 GB of memory is reserved for each vCPU, and at least 10 GB of idle disk space is available for installing the test tool.

Preparations

The following uses an ECS running CentOS 7.4 as an example.

1. Remotely log in to the ECS.
2. Install dependencies.

   yum install -y m4 numactl* automake bison bzip2
3. Upgrade GCC.

   wget https://benchmark-packages.oss-cn-qingdao.aliyuncs.com/gcc7.zip

   unzip gcc7.zip

   cd gcc7

   bash make_gcc.sh

   The installation takes about 40 minutes. Information similar to the following is displayed if the installation is successful:

   ```
   gcc version 7.3.0 (GCC)
   ```
4. Configure environment variables.
   a. Run the following command to configure the environment variables:

      vim /etc/profile

   b. Add the following information to the configuration file:

      ```
      export PATH=/usr/local/gcc/bin:$PATH
      export
      ```



```
LD_LIBRARY_PATH=/usr/local/gcc/lib64:/usr/local/gmp/lib:/usr/local/mpfr/lib:/usr/local/mpc/lib:$LD_LIBRARY_PATH
export MANPATH=/usr/local/gcc/share/man:$MANPATH
```

c. Run the following command for the environment variables to take effect:

source /etc/profile

d. Check the GCC version.

gcc -v

```
gcc version 7.3.0 (GCC)
```

5. Upgrade glibc.

In this example, glibc 2.27 is used, which must be installed in the GCC 7.3.0 environment. Otherwise, the system will display a message indicating an early GCC version.

a. Upload the configuration file to the **/home** directory and run the following commands to install glibc:

cd /home

tar -zxvf glibc-2.27.tar.gz

cd glibc-2.27

mkdir build

cd build

export LD_LIBRARY_PATH=/usr/local/mpc-1.0.3/lib:/usr/local/gmp-6.1.0/lib:/usr/local/mpfr-3.1.4/lib:/usr/local/gcc-7.3.0/lib:/usr/local/isl-0.18/lib:/

../configure --prefix=/usr/local/glibc-2.27 --disable-profile --enable-add-ons --with-headers=/usr/include --with-binutils=/usr/bin --disable-sanity-checks --disable-werror

make -j 40

make -j 40 install

b. Run the following command to configure the environment variables:

vim /etc/profile

c. Add the following information to the configuration file:

```
export PATH=/usr/local/glibc-2.27/bin:$PATH
```

d. Run the following command for the environment variables to take effect:

source /etc/profile

e. Check the glibc version.

ldd --version

```
ldd (GNU libc) 2.27
```

Downloading the Benchmark Suite

Download the SPEC CPU®2017 benchmark suite.

In this example, the 1.0.5 version is used. You are advised to use SPEC CPU®2017 of version 1.0.5 or later.

Installing the Benchmark Suite

This section describes how to install SPEC CPU®2017.

1. Upload the SPEC CPU®2017 installation package to the specified directory on the ECS.

   In this example, the installation directory is **./spec2017**.

2. Decompress **speccpu2017_config.zip** to the specified directory.

   In this example, the package is decompressed to **cpu2017/config/**.

   unzip speccpu2017_config.zip -d spec2017/config/

3. Go to the SPEC CPU®2017 installation directory and install the tool.

   cd ./spec2017

   ./install.sh

   When the system displays a message asking you to check the installation directory, ensure that the directory is correct and press **y**.

   Information similar to the following is displayed if the installation is successful:

   ```
   Installation successful.
   ```

Evaluating the Instance Compute Performance

1. Set environment variables.
   - Check the GCC and glibc versions.

     To do so, run the **gcc -v** and **ldd --version** commands, respectively.

     Ensure that the GCC version is 7.3.0 or later, and the glibc version is 2.27 or later.
   - Run the **source shrc** command to associate the environment variables and library files required by SPEC CPU®2017 with the **/spec2017** directory.
2. Modify the configuration file.

   Run the following commands to download the configuration file required for the test and copy the file to the **config** directory in the SPEC CPU®2017 installation folder:

   wget https://benchmark-packages.oss-cn-qingdao.aliyuncs.com/speccpu2017_config.zip

   unzip speccpu2017_config.zip -d /spec2017/config/

3. Run the following command to perform a multi-vCPU RateInt test:

   ./bin/runcpu --config=spec17-opti-gcc7.3.cfg --copies=`cat /proc/cpuinfo | grep process | wc -l` --loose intrate

4. Run the following command to delete the files generated during the test to release disk space:

   rm -fr benchspec/CPU/*/run/*

   By default, performing an intrate test consumes 3 GB to 4 GB of storage space in the disk. Therefore, delete the generated files after the test is complete.

5. Perform the test multiple times if you need to obtain released test data. After the preceding operations are performed, the test is performed at the performance base only once, and the data cannot be released.
   - To perform the test for multiple times, add **-n** in the command.

     ./bin/runcpu --config=spec17-opti-gcc7.3.cfg --copies=`cat /proc/cpuinfo | grep process | wc -l` --loose intrate -n 20
   - To test a single item:

     For example, to test only **502.gcc_r**, run the following command:

     ./bin/runcpu --config=spec17-opti-gcc7.3.cfg 502.gcc_r

Table 1-1 SPEC CPU®2017 test items

Rate Int	Rate FP	Speed Int	Speed FP
500.perlben_r	503.bwaves_r	600.perlbench_s	603.bwaves_s
502.gcc_r	507.cactuBSSN_r	602.gcc_s	607.cactuBSSN_s
505.mcf_r	508.namd_r	605.mcf_s	619.lbm_s
520.omnetpp_r	510.parest_r	620.omnetpp_s	621.wrf_s
523.xalancbmk_r	511.povray_r	623.xalancbmk_s	627.cam4_s
525.x264_r	519.lbm_r	625.x264_s	628.pop2_s
531.deepsjeng_r	521.wrf_r	631.deepsjeng_s	638.imagick_s
541.leela_r	526.blender_r	641.leela_s	644.nab_s
548.exchange2_r	527.cam4_r	648.exchange2_s	649.fotonik3d_s
557.xz_r	538.imagick_r	657.xz_s	654.roms_s
N/A	544.nab_r	N/A	N/A
N/A	549.fotonik3d_r	N/A	N/A
N/A	554.roms_r	N/A	N/A

Obtaining the Evaluation Result

- After the evaluation is complete, go to the **/spec2017/result/** directory to obtain evaluation files.

  CPU2017.xxx.intrate.refrate.txt, **CPU2017.xxx.fprate.refrate.txt**, **CPU2017.xxx.intspeed.refspeed.txt**, and **CPU2017.xxx.fpspeed.refspeed.txt**

- During the evaluation, if you want to obtain the result of the item for which the test is already complete, run the following command (value **ratio** is the result):

  less CPU2017.XXX.log.debug | grep Success

2 Compute Performance Appraisals for Mainstream Instances

Scenarios

The cloud platform provides multiple ECS types for different compute and storage capabilities. Each ECS type provides various flavors with different vCPU and memory configurations for you to select. This section uses general computing-plus C6s, general computing-plus C6, and memory-optimized M6 ECSs as examples to describe how to use the benchmark test program from Standard Performance Evaluation Corporation, Integer (SPECInt) to evaluate the compute performance of theses ECSs running Linux.

Description

The evaluation results are obtained by running SPEC CPU®2017 on the tested ECSs. The test is performed 10 times at the performance base. Then, the average score and standard deviation are obtained based on the 10 test results. Take SPECrate®2017 Integer as an example. During the test, the number of vCPUs of the tested ECS is the same as that on the copies. Concurrently run the copies at the performance base obtained using SPECrate®2017 Integer. A higher score indicates a higher throughput per unit time.

- For instructions about how to obtain the SPEC test tool, visit SPEC CPU®2017.
- For details about ECS flavors, see ECS Types.
- For instructions about how to evaluate the compute performance of an ECS, see 1 Methods of Evaluating Instance Compute Performance.
- The test result is updated every half a year.

The following table lists the test environment.

Table 2-1 Test environment

Parameter	Description
OS	Linux
Vendor	HUAWEI CLOUD
Compiler	GCC v7.3.0
SPEC benchmark suite	speccpu_2017 (version: 1.0.x)

Parameter	Description
SPEC CPU®2017 test program	SPECrate®2017 Integer
Image	CentOS 7.4

Evaluation Result

- **General computing-plus C6s**

Table 2-2 Evaluation result of general computing-plus C6s ECSs running Linux

Flavor	vCPUs	Memory (GB)	Tests	Average Rate	Standard Deviation
c6s.large.2	2	4	10	4.430	0.011
c6s.xlarge.2	4	8	10	8.507	0.027
c6s.2xlarge.2	8	16	10	16.499	0.047
c6s.3xlarge.2	12	24	10	24.294	0.062
c6s.4xlarge.2	16	32	10	32.015	0.109
c6s.6xlarge.2	24	48	10	46.841	0.089
c6s.8xlarge.2	32	64	10	60.650	0.137
c6s.12xlarge.2	48	96	10	93.291	0.190
c6s.16xlarge.2	64	128	10	120.540	0.379

- **General computing-plus C6**

Table 2-3 Evaluation result of general computing-plus C6 ECSs running Linux

Flavor	vCPUs	Memory (GB)	Tests	Average Rate	Standard Deviation
c6.large.2	2	4	10	7.315	0.01
c6.xlarge.2	4	8	10	14.09	0.02
c6.2xlarge.	8	16	10	26.394	0.11

Flavor	vCPUs	Memory (GB)	Tests	Average Rate	Standard Deviation
2					
c6.3xlarge.2	12	24	10	38.848	0.05
c6.4xlarge.2	16	32	10	50.417	0.058
c6.6xlarge.2	24	48	10	57.536	0.164
c6.8xlarge.2	32	64	10	66.247	0.212
c6.12xlarge.2	48	96	10	109.438	0.284
c6.16xlarge.2	64	128	10	124.416	0.403
c6.22xlarge.2	88	176	10	149.042	0.211

- **Memory-optimized M6**

Table 2-4 Evaluation result of memory-optimized M6 ECSs running Linux

Flavor	vCPUs	Memory (GB)	Tests	Average Rate	Standard Deviation
m6.large.2	2	4	10	7.315	0.01
m6.xlarge.2	4	8	10	14.09	0.02
m6.2xlarge.2	8	16	10	26.394	0.11
m6.3xlarge.2	12	24	10	38.848	0.05
m6.4xlarge.2	16	32	10	50.417	0.058
m6.6xlarge.2	24	48	10	57.536	0.164
m6.8xlarge.2	32	64	10	66.247	0.212
m6.16xlarge.2	64	128	10	124.416	0.403

3 Change History

Released On	Description
2020-12-17	This issue is the first official release.

