

A decorative graphic on the left side of the page, consisting of a network of interconnected nodes and lines, resembling a mesh or a web, rendered in a lighter blue color against the dark blue background.

# Multi-MN HA Tutorial

Version: ZStack Cloud 4.3.12

Issue: V4.3.12

# Copyright Statement

---

Copyright © 2022 Shanghai Yunzhou Information and Technology Ltd. All rights reserved.

Without prior written consent of Shanghai Yunzhou Information and Technology Ltd., any organization and any individual do not have the right to extract, copy any part of or all of, and are prohibited to disseminate the contents of this document in any manner.

## Trademark

Shanghai Yunzhou Information and Technology Ltd. reserves all rights to its trademarks, including , but not limited to ZStack and other trademarks in connection with Shanghai Yunzhou Information and Technology Ltd.

Other trademarks or registered trademarks presented in this document are owned or controlled solely by its proprietaries.

## Notice

The products, services, or features that you purchased are all subject to the commercial contract and terms of Shanghai Yunzhou Information and Technology Ltd., but any part or all of the foregoing displayed in this document may not be in the scope of your purchase or use. Unless there are additional conventions, Shanghai Yunzhou Information and Technology Ltd. will disclaim any statement or warranty, whether implicit or explicit, on the contents of this document.

In an event of product version upgrades or other reasons, the contents of this document will be irregularly updated and released. Unless there are additional conventions, this document, considered solely as a reference guide, will not make any warranty, whether implicit or explicit, on all the statements, information, or suggestions.

# Contents

---

<b>Copyright Statement.....</b>	<b>I</b>
<b>1 Installation and Deployment.....</b>	<b>1</b>
1.1 Overview.....	1
1.2 Installation and Deployment.....	1
1.2.1 Prepare Software Tools.....	2
1.2.2 Check Hardware Devices.....	2
1.2.3 Check Network Connections.....	3
1.2.4 Install the Operating System.....	4
1.2.5 Configure Networks.....	8
1.2.5.1 Configure the Management Network.....	9
1.2.5.2 Configure the VM Data Network.....	10
1.2.6 Install the HA Suite.....	11
1.2.6.1 Use the Command Line.....	11
1.2.6.2 Configuration Files.....	16
1.2.7 Install the License.....	20
1.3 Upgrade the Cluster.....	20
1.4 Other Operations.....	23
1.4.1 Monitoring Alarm.....	23
1.4.2 Log Output.....	24
<b>2 HA Test and Recovery.....</b>	<b>25</b>
2.1 Planned O&M.....	25
2.1.1 Single-MN Maintenance.....	25
2.1.2 Dual-MN Maintenance.....	26
2.2 MN Troubleshooting.....	27
2.2.1 Single-MN Troubleshooting.....	27
2.2.2 Dual-MN Troubleshooting.....	27
2.2.3 MN Database Backup and Recovery.....	28
<b>3 CLI Guidance.....</b>	<b>30</b>
3.1 Introduction.....	30
3.2 -h.....	30
3.3 version.....	30
3.4 install-ha.....	31
3.5 stop-node.....	36
3.6 start-node.....	36
3.7 upgrade-mn.....	36
3.8 upgrade-ha.....	37
3.9 demote.....	38
3.10 status.....	38
3.11 show-config.....	39
3.12 sample-config.....	39

3.13 collect-log.....40

**Glossary..... 41**

# 1 Installation and Deployment

## 1.1 Overview

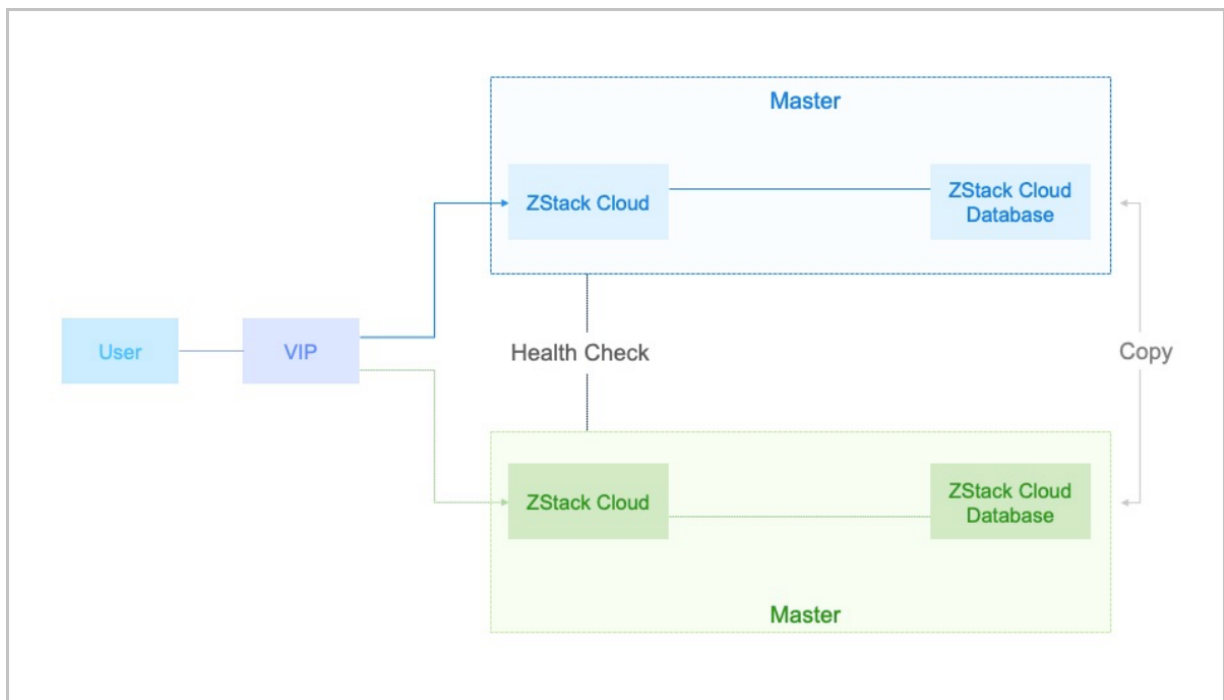
ZStack Cloud provides high availability (HA) for multiple management nodes (MNs, also known as hosts) by using separate HA suites. When any one of the MNs is disconnected, the HA will be triggered in seconds to ensure the business continuity.

This Tutorial provides an example of a dual-MN HA scenario.

### HA Mechanism

In a dual-MN model, each MN runs a **zsha2** HA process. This process monitors the critical services, including the MN service, UI service, and database service, on each MN in real time. When any critical service is down, the Cloud immediately triggers virtual IP (VIP) migration through Keepalived, and then attempts to restore the downtime service.

**Figure 1-1: Dual-MN HA**



## 1.2 Installation and Deployment

This section mainly describes how to install and deploy a dual-MN HA environment.

## 1.2.1 Prepare Software Tools

The administrator needs to prepare the following necessary software packages to facilitate the installation and deployment process.

- ZStack Cloud custom ISO (c76 is recommended.)
  - ZStack Cloud custom ISO:
    - C76: ZStack-Cloud-x86\_64-DVD-4.3.12-c76.iso
    - C74: ZStack-Cloud-x86\_64-DVD-4.3.12-c74.iso
    - Download address: [Click here](#)
  - ZStack Cloud installation package:
    - Software: ZStack-Cloud-installer-4.3.12.bin
    - Download address: [Click here](#)
- Multi-MN HA suite
  - Software: ZStack-Multinode-HA-Suite-4.3.12.tar.gz
  - Download address: [Click here](#)



### Note:

After you download the required software, confirm the integrity of the file by using the MD5 checksum tool.

## 1.2.2 Check Hardware Devices

This topic provides a scenario where two x86 servers are used to deploy a dual-MN HA environment. The configuration information is listed in [Table 1-1: Server Configuration](#). The administrator can adjust the capacity ratio of the CPU, memory, and hardware according to actual business requirements.

**Table 1-1: Server Configuration**

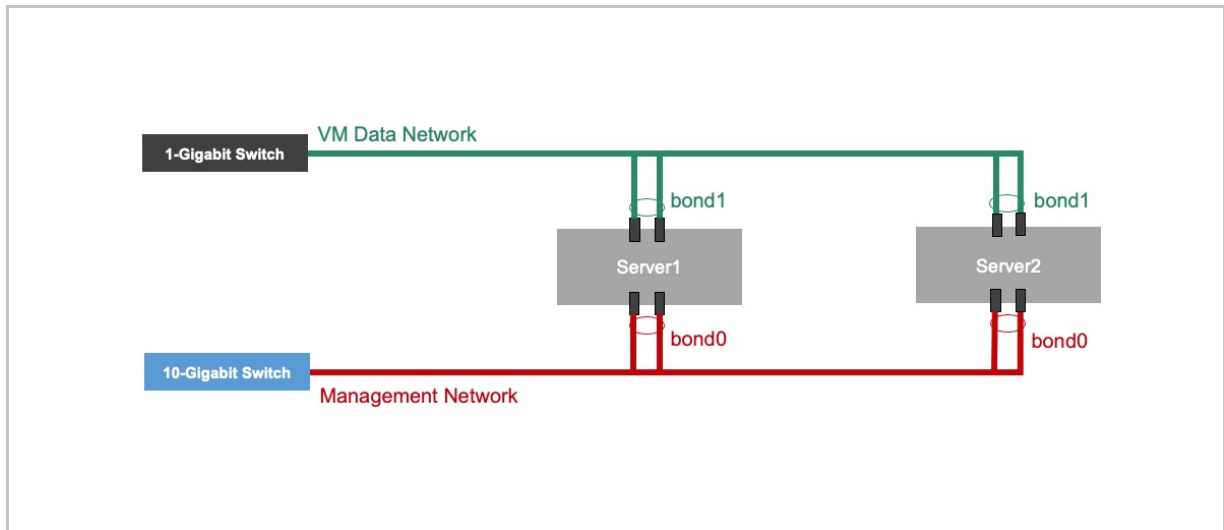
	Accessory	Model	Quantity	Total
Server	CPU	Intel(R) Xeon(R) CPU E5-2630 v4 @ 2.20GHz	2	2
	Memory	DDR4 16GB	8	
	Motherboard	Two-way server standard motherboard	1	

RAID controller	Supports SAS/SATA RAID 0/1/10 Supports the passthrough mode	1
SSD	Intel SSD DC S3610 480GB	2
HDD1	SAS HDD 300GB 3.5", 15K RPM	2
HDD2	NL SAS HDD 2TB 3.5", 7.2K RPM	6
1-Gigabit Ethernet port	1GbE, RJ45	2
10-Gigabit Ethernet port	10GbE, SFP+	2
Optical module	-	
HBA card	-	
Remote management	DELL iDRAC Enterprise	1
Power supply	1100W standard power supply	2

In addition, this scenario involves one 10 Gigabit switch, one 1-Gigabit switch, and several cat5 jumpers.

### 1.2.3 Check Network Connections

The administrator puts the servers and network devices mentioned before in place, and connects them to check the network connections according to the network topology.

**Figure 1-2: Network Topology**

## 1.2.4 Install the Operating System

### Procedure

#### 1. Preparations

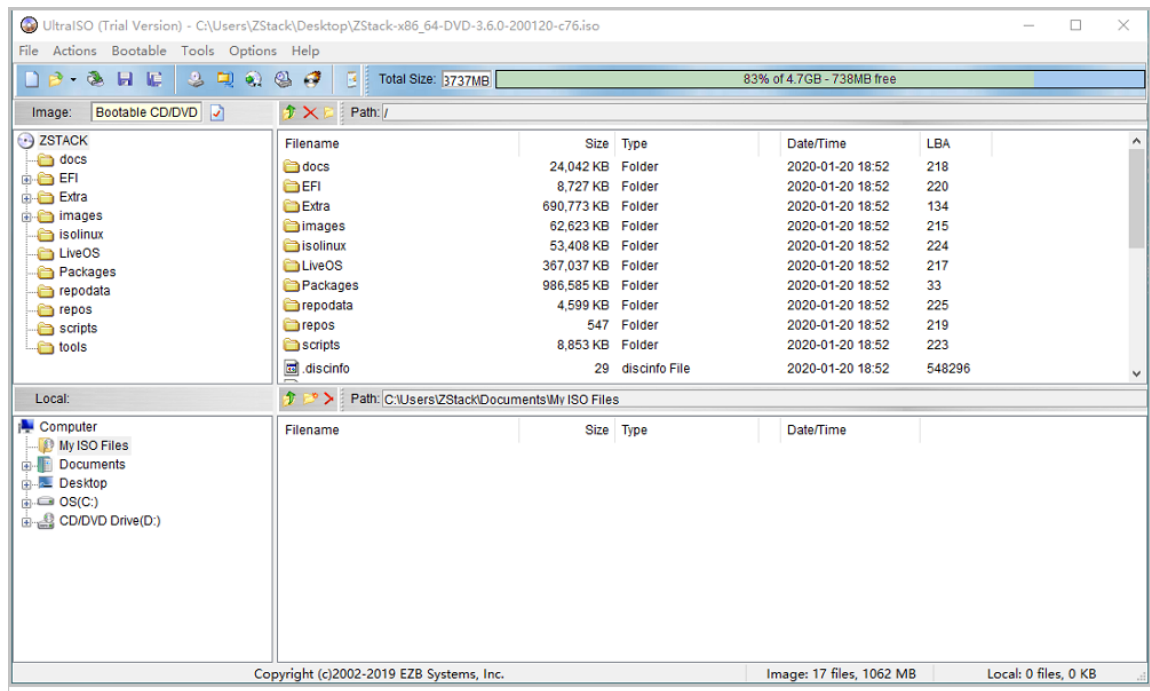
The administrator powers on the network devices and servers, manually starts the server to enter the BIOS mode, and makes the following preparations:

- Activate all CPU cores, enable the hyper-threading function, and set the system performance to the highest performance state.
- Turn on hardware virtualization to accelerate the optimization of hardware virtualization.
- Go to the RAID controller settings, configure RAID1 (Mirror) for the two system hard drives, and set the passthrough mode for the remaining hard drives.

#### 2. In UltraISO, open the DVD image of ZStack Cloud.

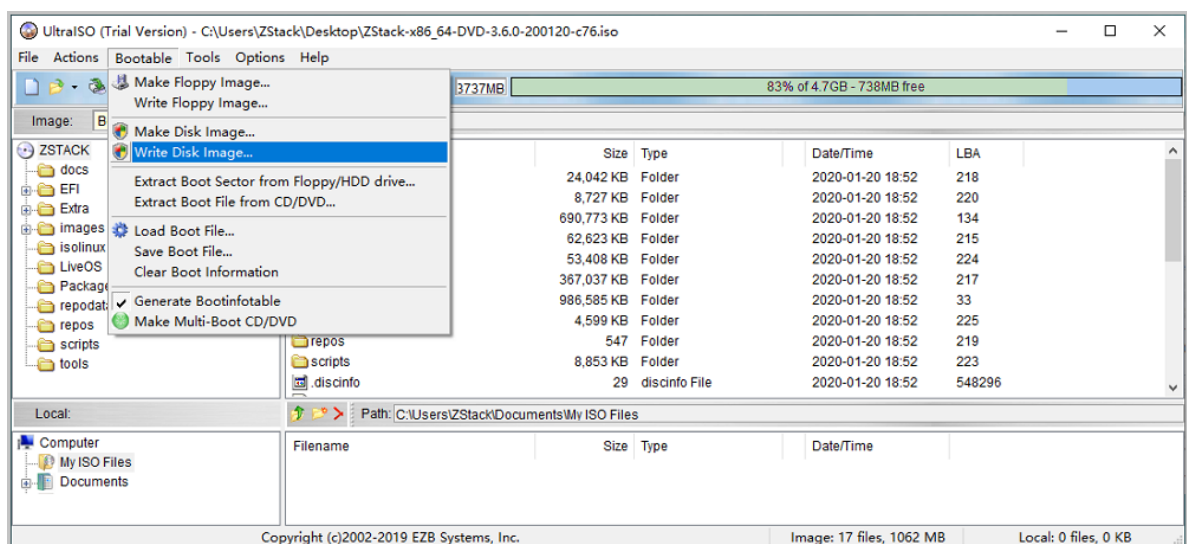
- You can burn the ISO image of ZStack Cloud operating system to an installation disk by using a DVD-RW device. You can also burn the ISO file to the USB drive by using the UltraISO tool.
- Open UltraISO, click the **File** button, and select the ISO file that you downloaded before.



**Figure 1-3: Open DVD Image in UltraISO**

3. Write the DVD image to the USB drive.

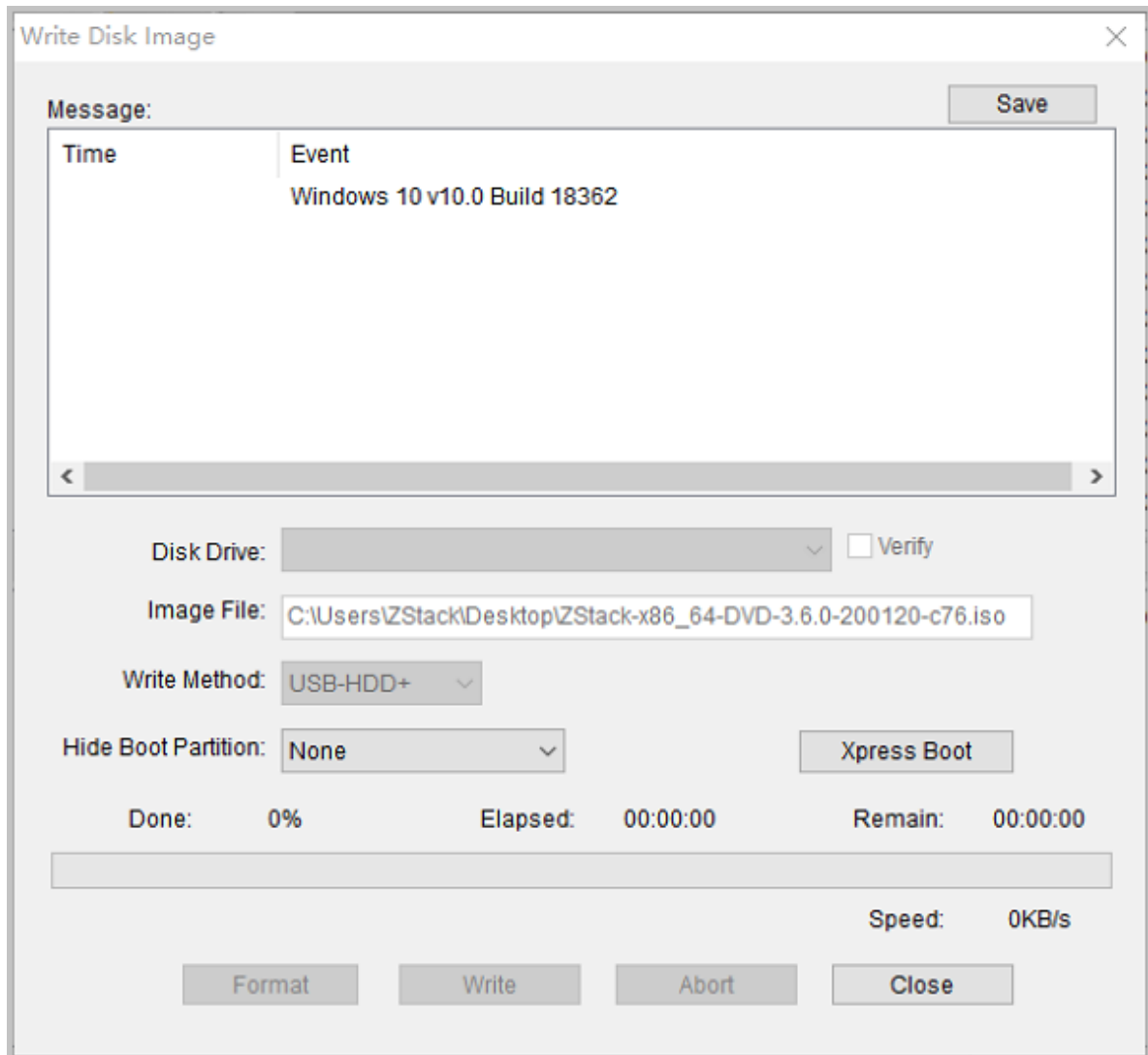
In UltraISO, choose **Bootable > Write Disk Image**.

**Figure 1-4: Write DVD Image in UltraISO**

4. Verify that the DVD image of ZStack Cloud is written to UltraISO.

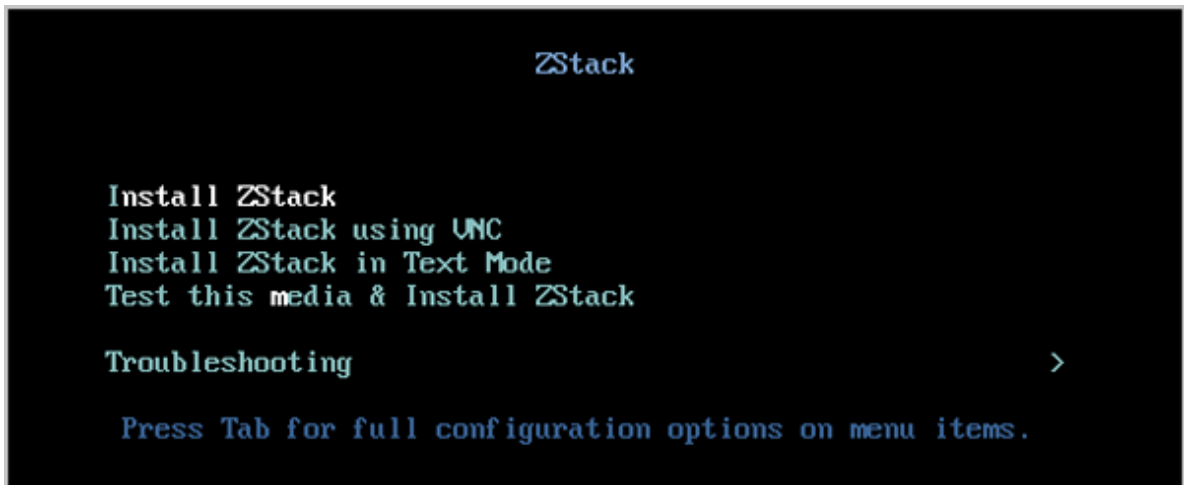
- If the system has only one USB drive plugged in, the USB drive will be burned and written by default. Before you burn the image to the USB drive, **make sure that the USB drive is backed up.**
- For other options, use the default settings. After the settings are completed, click **Write**.

**Figure 1-5: Verify the Writing of ISO Image in UltraISO**



**5. Enter the installation guide page.**

The ISO image is burned to the USB drive. Then, this USB drive can be used as a boot disk, which supports the Legacy and UEFI boot modes. Then, the administrator boots the node through the installation media and enters the installation guide page.

**Figure 1-6: USB Drive Guide Page**

**6. Install the operating system.**

By default, the operating system installation begins.

On the installation summary page, some options are preconfigured as below. You can modify the configurations as needed.

- **DATE & TIME:** UTC/GMT+8
- **LANGUAGE SUPPORT:** English (United States)
- **KEYBOARD:** English (United States)

Generally, the administrator does not need to modify the configurations. However, the administrator needs to partition the hard disks. The recommended partitioning method is as follows (UEFI mode):

- */boot/efi*: create a 500 MB partition
- */boot*: create a 1 GB partition
- *swap* (exchange partition): create a 32 GB partition
- */* (root partition): configure the remaining capacity

After the partitions are configured, click **SOFTWARE SELECTION**. On the software selection page, select the compute node installation, and go back to the installation summary page.

Click **Begin Installation**. Then, the installation process proceeds automatically. The administrator needs to set the root account and password.

After the installation is completed, reboot the server and unplug the USB drive. If the installation is successful, the server restarts and enters the operating system login prompt. You can use your root account and password to log in to the operating system.



**Note:**

The administrator can change the password as needed.

## 1.2.5 Configure Networks

The administrator can configure networks after installing the operating system on both servers. ZStack Cloud provides a convenient network configuration script in the `/usr/local/bin/` directory. The administrator can use this script to quickly configure the interface and bridge information.

The following table lists the network information of the two MNs and the VIP settings used for Keepalived communication in this scenario.

**Table 1-2: Management Network**

Server	NIC1	NIC2	Bond	Bridge	IP Address	Netmask	Gateway
MN1	eth0	eth1	bond0	br_bond0	192.168.195.200	255.255.0.0	192.168.0.1
MN2	eth0	eth1	bond0	br_bond0	192.168.196.125	255.255.0.0	192.168.0.1

**Table 1-3: VM Data Network**

Node	NIC1	NIC2	Bond	Bridge	IP Address	Netmask	Gateway
MN1	em1	em2	bond1	-	-	-	-
MN2	em1	em2	bond1	-	-	-	-

**Table 1-4: VIP**

-	IP Address	Netmask
VIP	192.168.199.151	255.255.0.0

**Note:**

The virtual IP address (VIP) is used to log in to the UI. Do not use the VIP to log in to a MN through SSH.

- The data provided above is sample data. The administrator can change the data in the actual deployment environment.
- The gateway must be provided by a physical network device. Meanwhile, **the gateway is used to detect the network status.**

In the following sections, we will describe how to configure the management network and the VM data network, respectively.

### 1.2.5.1 Configure the Management Network

Assume that the settings of the management network is as follows.

**Table 1-5: Management Network**

Server	NIC1	NIC2	Bond	Bridge	IP Address	Netmask	Gateway
MN1	eth0	eth1	bond0	br_bond0	192.168.195.200	255.255.0.0	192.168.0.1
MN2	eth0	eth1	bond0	br_bond0	192.168.196.125	255.255.0.0	192.168.0.1

On **MN1**, run the following commands:

```
# Create bond0.
[root@localhost ~]# zs-bond-lacp -c bond0

# Attach eth0 and eth1 to bond0.
[root@localhost ~]# zs-nic-to-bond -a bond0 eth0
[root@localhost ~]# zs-nic-to-bond -a bond0 eth1

# After the preceding bonds are configured, configure LACP aggregation
  on the corresponding switch port.

# Create br_bond0, and configure the IP address, netmask, and gateway.
[root@localhost ~]# zs-network-setting -b bond0 192.168.195.200 255.
255.0.0 192.168.0.1

# Check whether bond0 is successfully created.
[root@localhost ~]# zs-show-network
...
```

```
-----
| Bond Name | SLAVE(s) | BONDING_OPTS
```

```

| bond0      | eth0      | miimon=100 mode=4 xmit_hash_policy=
layer2+3    | |
|           | eth1      |
|           | |

```

On **MN2**, run the similar commands.



#### Note:

- After you attach eth0 and eth1 to bond0, you must configure LACP aggregation for the port of the corresponding switch. Otherwise, the network communication might be abnormal. If the switch does not support LACP aggregation, contact the network device manufacturer to replace the device.
- After you create a bridge named br\_bond0 via bond0, the bridge will provide management network services.
- For the IP address, netmask, and gateway of the bridge, you can enter a value according to your actual needs.
- After the management network is configured, you can check the configuration by using the **ping** command. If the configuration is correct, the management IP addresses of these two MNs can ping each other.
- We recommend that you use a 10-Gigabit or above bandwidth for the management network. If the management network is deployed independently, a 1-Gigabit bandwidth is allowed.

After the management network is configured, you can configure the VM data network accordingly.

### 1.2.5.2 Configure the VM Data Network

Assume that the settings of the VM data network is as follows.

**Table 1-6: VM Data Network**

Node	NIC1	NIC2	Bond	Bridge	IP Address	Netmask	Gateway
MN1	em1	em2	bond1	-	-	-	-
MN2	em1	em2	bond1	-	-	-	-

On **MN1**, run the following commands:

```

# Create bond1.
[root@localhost ~]# zs-bond-lacp -c bond1

```

```
# Attach em1 and em2 to bond1.
[root@localhost ~]# zs-nic-to-bond -a bond1 em1
[root@localhost ~]# zs-nic-to-bond -a bond1 em2

# After the preceding bonds are configured, configure LACP aggregation
on the corresponding switch port.

# You do not need to create a bridge for the VM data network.

# Check whether bond1 is successfully created.
[root@localhost ~]# zs-show-network
...
-----
| Bond Name | SLAVE(s) | BONDING_OPTS
-----
| bond1     | em1      | miimon=100 mode=4 xmit_hash_policy=
layer2+3    |          |
|           | em2      |
-----
```

On **MN2**, run the similar commands.



#### Note:

After you attach em1 and em2 to bond1, you must configure LACP aggregation for the port of the corresponding switch. Otherwise, the network communication might be abnormal. If the switch does not support LACP aggregation, contact the network device manufacturer to replace the device.

## 1.2.6 Install the HA Suite

This section introduces two methods to install the HA suite.

- Install the HA suite directly by using the command line.
- Install the HA suite by writing configuration files.



#### Note:

With the same configurations, the command line method has higher priority than the configuration file method.

### 1.2.6.1 Use the Command Line

#### Context

The administrator installed two ZStack Cloud MNs with the latest version, and installed licenses for these two MNs. Now, the administrator wants to install a **multi-MN HA suite** on one of the MNs to achieve HA.

- MN1 (192.168.195.200)
- MN2 (192.168.196.125)

Assume that the administrator wants to install the HA suite on MN1. In this case, MN1 is the active MN, and MN2 is the standby MN.

## Procedure

### 1. Import the HA suite.

Run the following commands to import the HA suite to MN1 and decompress the suite:

```
# Copy the HA suite to MN1 by using the scp tool.
[root@localhost ~]# ls
ZStack-Multinode-HA-Suite-4.3.12.tar.gz

# Decompress the suite. Then, two executable files are generated:
  zsha2 and zstack-hamon.
[root@localhost ~]# tar zxvf ZStack-Multinode-HA-Suite-4.3.12.tar.gz
zsha2 //A program for installing and managing multi-node HA.
zstack-hamon //A program for monitoring multi-node HA.
```

### 2. Initialize the HA suite.

On MN1, run the following commands to install the HA suite:

```
[root@localhost ~]# chmod +x zsha2 zstack-hamon
[root@localhost ~]# ./zsha2 install-ha -nic br_bond0 -gateway 192.
168.0.1 -slave "root:password@192.168.196.125" \
-vip 192.168.199.151 -myip 192.168.195.200 -db-root-pw zstack.mysql.
password -time-server 192.168.196.125 -cidr 192.168.0.0/16 -yes
```



#### Note:

- After the installation commands are executed, the database of the active and backup MNs will be automatically backed up, and then the suite will be installed.
- To install the HA suite, you need to put **zsha2** and **zstack-hamon** in the same directory. During the installation process, **zsha2** will automatically deploy **zstack-hamon** and related configuration files.
- Description of parameters in the installation commands:
  - **-nic**: The name of the physical device. This parameter is used to configure a VIP. In production environments, this parameter is generally a management network bridge, such as **-nic br\_bond0**.
  - **-gateway**: The arbitration gateway of the active and backup MNs, for example, **-gateway 192.168.0.1**.



- `-slave`: Specifies the standby MN, for example, `-slave "root:password@192.168.196.125"`.

**Note:**

- During the installation process, the database of the standby MN will be overwritten by that of the active MN. Please exercise caution.
- When you install an HA suite, we recommend that you set the root password to a regularly used password for easy and quick deployment. You can change the root password later as needed. The HA suite will no longer depend on the system root password.
- If the root password contains special shell characters, for example, ' " \* ? \ ~ ` ! # \$ & |, enter \ to escape these characters.

For example, if the system password is ' " \* ? \ ~ ` ! # \$ & |, you can escape the password as follows:

```
-slave "root:\' \" \* \? \\ \~ \` \! \# \$ \% \|@192.168.196.125"
```

- `-vip`: Specifies the VIP for Keepalived communication, for example, `-vip 192.168.199.151`.
- `-myip`: Optional. Specifies the local IP address, for example, `-myip 192.168.195.200`.
- `-db-root-pw`: The database root password of the active and standby MNs, for example, `-db-root-pw zstack.mysql.password`. Make sure that both MNs share the same root password.
- `-time-server`: Specifies a time synchronization server for unified time synchronization, for example, `./zsha2 install-ha -time-server 192.168.196.125`.

**Note:**

You can specify multiple time servers at a time, for example, `./zsha2 install-ha -time-server 192.168.196.125,192.168.196.126`.

- `-cidr`: Optional. Specifies the network range, which must cover the IP address, VIP, and gateway of the active and standby MNs. For example, `./zsha2 install-ha -cidr 192.168.0.0/16`.

**Note:**

If not specified, the system will automatically calculate a minimum network range, which might fail to meet the requirements. We recommend that you specify a network range.

- **-force:** Optional. Force runs the **zsha2** installation command in the active MN when the database of the active and standby MNs cannot be automatically synchronized for a long time. For example, `./zsha2 install-ha -force`.

**Note:**

We recommend that you back up both databases before performing a force installation.

- **-repo:** Optional. Specifies the YUM repository, which defaults to the local repository, for example, `./zsha2 install-ha -repo zstack-local`.
- **-timeout:** Optional. The timeout for copying the databases of the active and backup MNs. Default value: 600. Unit: second. For example, `./zsha2 install-ha -timeout 600`.
- **-yes:** Optional. Indicates that all settings are allowed.

After the HA suite is initialized, you can run the following commands to view the status of the MNs:

```
# View the status of MN1.
[root@localhost ~]# zsha2 status
Status report from 192.168.195.200
=====
Owns virtual address:          yes //MN1 has obtained a VIP. Only
one MN can obtain VIP at the same time.
Self 192.168.195.200 reachable: yes //MN1 is reachable.
Gateway 192.168.0.1 reachable: yes //The current gateway is
reachable.
VIP 192.168.199.151 reachable:  yes //The VIP is reachable.
Peer 192.168.196.125 reachable:  yes //MN2 is reachable.
Keepalived status:            active //The Keepalived service is
in the running state.
ZStack HA Monitor:            active //The HA monitoring service
isin the running state.
MySQL status:                  mysqld is alive //The database is
running properly.
MN status: Running [PID:6500] //The MN is running properly.
UI status: Running [PID:9785] http://192.168.195.200:5000 //The UI
is running properly.

Slave Status:
-----
      Slave_IO_Running: Yes //Slave I/O is running properly.
      Slave_SQL_Running: Yes //Slave SQL is running properly.
      Last_Error:
```

```

        Seconds_Behind_Master: 0
            Last_IO_Error:
            Last_SQL_Error:

Warning: Permanently added '192.168.196.125' (ECDSA) to the list of
known hosts.
Status report from 192.168.196.125 //View the status of MN2.
=====
Owns virtual address:          no
Self 192.168.196.125 reachable: yes
Gateway 192.168.0.1 reachable: yes
VIP 192.168.199.151 reachable: yes
Peer 192.168.195.200 reachable: yes
Keepalived status:            active
ZStack HA Monitor:            active
MySQL status:                 mysqld is alive

Slave Status:
-----
        Slave_IO_Running: Yes
        Slave_SQL_Running: Yes
            Last_Error:
        Seconds_Behind_Master: 0
            Last_IO_Error:
            Last_SQL_Error:

Note: visit ZStack UI with http://192.168.199.151:5000

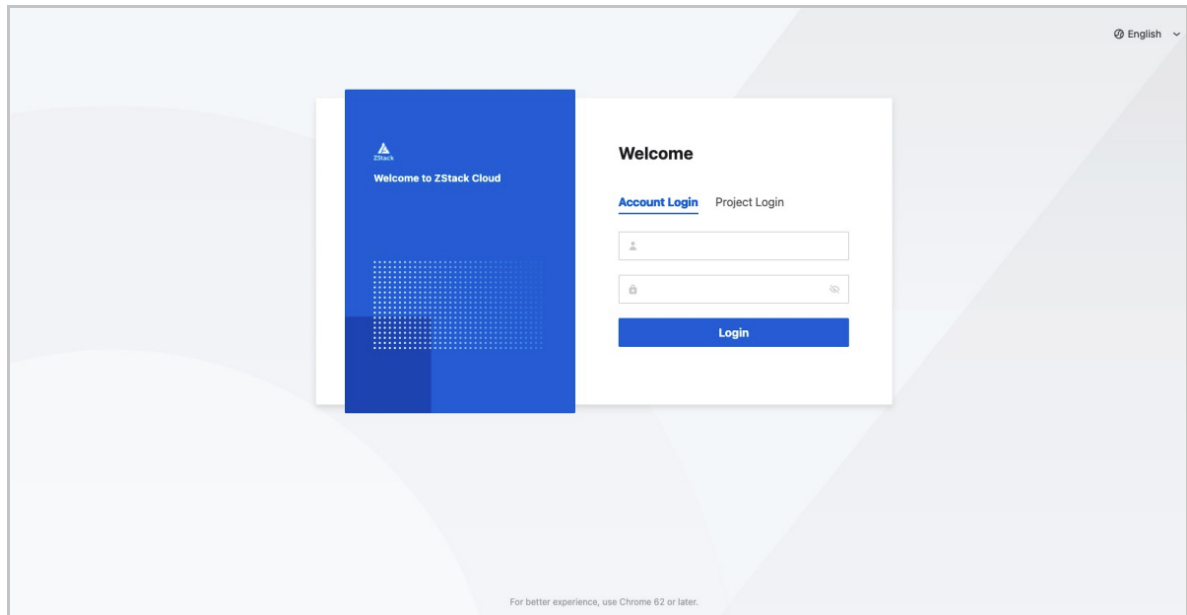
```

**Note:**

During the installation of the HA suite, SSH password-free login was automatically configured for these two MNs.

**3. Initialize the Cloud.**

The administrator can access the UI of MN1 through the VIP (192.168.199.151), and then complete the Cloud initialization.

**Figure 1-7: Login Page**

Run the following command on MN1. Then, MN1 switches to the backup MN online, and MN2 obtains the VIP (192.168.199.151) and becomes the active MN.

```
[root@localhost ~]# zsha2 demote
```

The administrator can refresh the UI (<http://192.168.199.151:5000>) of MN2 by using the VIP and complete the Cloud initialization.

## 1.2.6.2 Configuration Files

### Context

The administrator installed two ZStack Cloud MNs with the latest version, and installed licenses for these two MNs. Now, the administrator wants to install a multi-MN HA suite in one of the MNs to achieve HA.

- MN1 (192.168.195.200)
- MN2 (192.168.196.125)

Assume that the administrator wants to install the HA suite on MN1. In this case, MN1 is the active MN, and MN2 is the standby MN.

### Procedure

1. Import the HA suite.

Run the following commands to import the HA suite to MN1 and decompress the suite:

```
# Copy the HA suite to MN1 by using the scp tool.
[root@localhost ~]# ls
ZStack-Multinode-HA-Suite-4.3.12.tar.gz

# Decompress the suite. Then, two executable files are generated:
zsha2 and zstack-hamon.
[root@localhost ~]# tar zxvf ZStack-Multinode-HA-Suite-4.3.12.tar.gz
zsha2 //A program for installing and managing multi-node HA.
zstack-hamon //A program for monitoring multi-node HA.
```

## 2. Write the configurations.

Run the following commands to write the initialization configuration file for the HA suite:

```
[root@localhost ~]# chmod +x zsha2 zstack-hamon
[root@localhost ~]# ./zsha2 sample-config > zs-install.config
[root@localhost ~]# cat zs-install.config
{
  "gateway": "192.168.0.1", //The arbitration gateway of the active
and standby MNs.
  "virtualIp": "192.168.199.151", //Specifies the VIP for Keepalived
communications.
  "myIp": "192.168.195.200", //Specifies the local IP address.
  "peerIp": "192.168.196.125", //Specifies the IP address of the
peer MN.
  "peerSshUser": "root", //Specifies the SSH username of the peer MN
.
  "peerSshPass": "password", //Specifies the SSH password of the
peer MN.
  "peerSshPort": 22, //Specifies the SSH port of the peer MN.
  "dbRootPass": "zstack.mysql.password", //Specifies the root
password of the active and backup MNs. Make sure that both MNs share
the same root password.
  "interface": "br_bond0", //The name of the physical device. This
parameter is used to configure a VIP. In production environments,
this parameter is generally a management network bridge.
  "timeServer": "192.168.196.125" //Specifies a time synchronization
server for unified time synchronization.
}
```

The administrator needs to modify the parameters above according to specific deployment scenarios.

## 3. Initialize the HA suite.

Run the following command to initialize and install the HA suite:

```
[root@localhost ~]# ./zsha2 install-ha -config zs-install.config
```



**Note:**

- After the installation command is executed, the database of the active and backup MNs will be automatically backed up, and then the suite will be installed.
- To install the HA suite, you need to put **zsha2** and **zstack-hamon** in the same directory. During the installation process, **zsha2** will automatically deploy **zstack-hamon** and related configuration files.
- Description of parameters in the installation commands:  
 — **-config**: Optional. Installs the HA suite by initializing the configuration files.

After the HA suite is initialized, you can run the following commands to view the status of the MNs:

```
# View the status of MN1.
[root@localhost ~]# zsha2 status
Status report from 192.168.195.200
=====
Owns virtual address:          yes //MN1 has obtained a VIP. Only
one management node can obtain VIP at the same time.
Self 192.168.195.200 reachable: yes //MN1 is reachable.
Gateway 192.168.0.1 reachable: yes //The current gateway is
reachable.
VIP 192.168.199.151 reachable: yes //The VIP is reachable.
Peer 192.168.196.125 reachable: yes //MN2 is reachable.
Keepalived status:            active //The Keepalived service is
in the running state.
ZStack HA Monitor:            active //The HA monitoring service
is in the running state.
MySQL status:                  mysqld is alive //The database is
running properly.
MN status: Running [PID:6500] //The MN is running properly.
UI status: Running [PID:9785] http://192.168.195.200:5000 //The UI
is running properly.

Slave Status:
-----
      Slave_IO_Running: Yes //Slave I/O is running properly.
      Slave_SQL_Running: Yes //Slave SQL is running properly.
      Last_Error:
      Seconds_Behind_Master: 0
      Last_IO_Error:
      Last_SQL_Error:

Warning: Permanently added '192.168.196.125' (ECDSA) to the list of
known hosts.
Status report from 192.168.196.125 //View the status of MN2.
=====
Owns virtual address:          no
Self 192.168.196.125 reachable: yes
Gateway 192.168.0.1 reachable: yes
VIP 192.168.199.151 reachable: yes
Peer 192.168.195.200 reachable: yes
Keepalived status:            active
ZStack HA Monitor:            active
MySQL status:                  mysqld is alive

Slave Status:
```

```

-----
Slave_IO_Running: Yes
Slave_SQL_Running: Yes
Last_Error:
Seconds_Behind_Master: 0
Last_IO_Error:
Last_SQL_Error:

```

Note: visit ZStack UI with <http://192.168.199.151:5000>



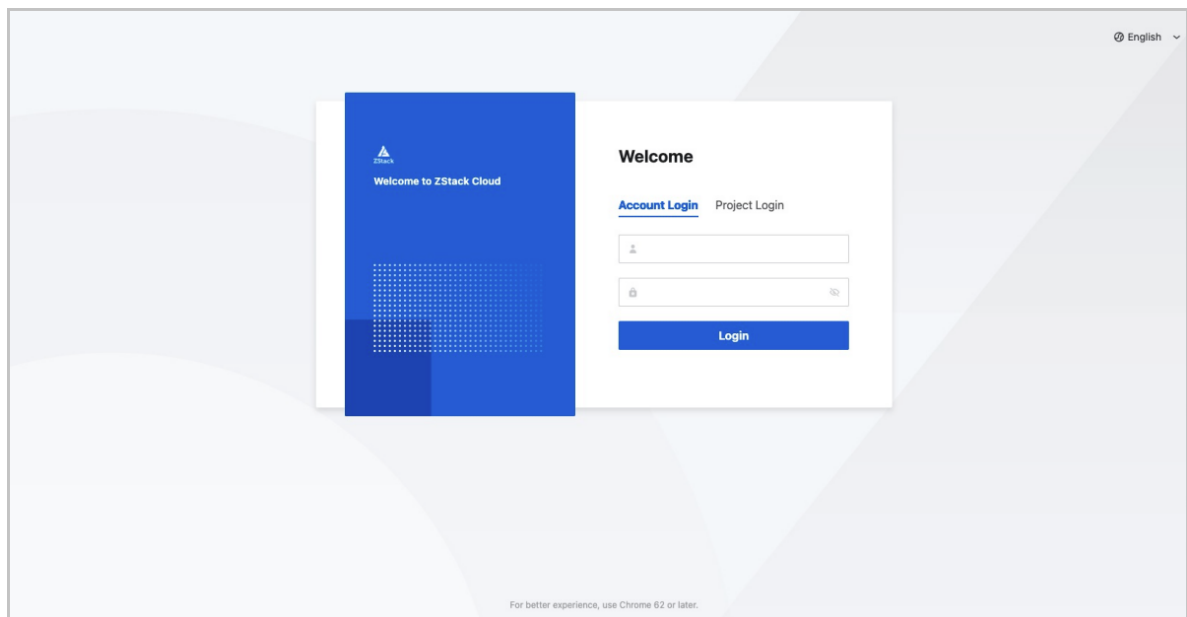
**Note:**

During the installation of the HA suite, SSH password-free login was automatically configured for these two MNs.

#### 4. Initialize the Cloud.

The administrator can access the UI (<http://192.168.199.151:5000>) of MN1 through the VIP (192.168.199.151), and then complete the Cloud initialization.

**Figure 1-8: Login Page**



Run the following command on MN1. Then, MN1 switches to the backup MN online, and MN2 obtains the VIP (192.168.199.151) and becomes the active MN.

```
[root@localhost ~]# zsha2 demote
```

The administrator can refresh the UI (<http://192.168.199.151:5000>) of MN2 by using the VIP and complete the Cloud initialization.

## 1.2.7 Install the License

In this scenario, the license type of two ZStack Cloud MNs must be the same.

The administrator can install licenses either via the UI or by using CLI.

### UI Method

1. Access the UI (<http://VIP:5000>) by using a VIP and log in to the Cloud as an admin.
2. Go to the **License Management** page and click **Upload license** in the upper right corner.  
On the displayed **Upload License** page, upload or drag-and-drop the dual-MN license you obtained before.

### CLI Method

The administrator can import licenses to these two MNs by using CLI.

## 1.3 Upgrade the Cluster

This topic provides an example of how to upgrade a dual-MN HA environment in ZStack Cloud.

The upgrade involves the following three steps:

1. Complete the preparations before the upgrade.
2. Upgrade the HA suite.
3. Upgrade the MNs.

Before you perform any upgrades, complete the following preparations:

1. Disable the VM HA functionality globally to avoid accidentally triggering VM HA and affecting the upgrade. Method: On the main menu of the MN UI, choose **Settings > Global Setting**. On the **Global Setting** page, disable the **VM HA** option. You can manually enable this functionality after the upgrade is completed.
2. Run the following command on each MN to back up your database:

```
[root@localhost ~]#zstack-ctl dump_mysql --file-name zstack-db-backup
```

3. Make sure that the zstack-upgrade script, the corresponding ISO, the installation and upgrade package, and the HA suite are downloaded for these two MNs.
4. Run the following command on each MN to update the local repo by using the latest ISO you downloaded before:

```
[root@localhost ~]#cd /root/
```



```
#bash /root/zstack-upgrade -r ZStack-Cloud-x86_64-DVD-4.3.12-c76.iso
```

After you obtain the new HA suite, you can use the suite to upgrade the existing **zsha2** service.

The steps are as follows:

1. Run the following command on any MN to check on which node the HA VIP is located. After you execute **zsha2**, the node whose "Owns virtual address" is yes in the returned result is the MN where the VIP is located.

```
[root@localhost ~]#zsha2 status
```

2. Log in to the MN where the VIP is located through IPMI, and run the following command to decompress the dual-MN HA suite:

```
[root@localhost ~]#tar zxvf ZStack-Multinode-HA-Suite-4.3.12.tar.gz
```

3. Run the following command to grant executable permissions to the decompressed **zsha2** and **zstack-hamon**:

```
[root@localhost ~]#chmod +x zsha2 zstack-hamon
```

4. Run the following command in the MN where the VIP is located to complete the upgrade of the HA suite:

```
[root@localhost ~]# ./zsha2 upgrade-ha
```

After you upgrade the HA suite, you can upgrade the MNs by following these steps:

1. Make sure that:

- The VIP is reachable.
- The current gateway is reachable.
- The standby MN is reachable.
- The database is synchronized.

You can view the MN status by running **zsha2 status**.

```
[root@localhost ~]# zsha2 status
Status report from 192.168.195.200
=====
Owns virtual address:          yes //MN1 has obtained a VIP. Only
one MN can obtain VIP at the same time.
Self 192.168.195.200 reachable: yes //MN1 is reachable.
Gateway 192.168.0.1 reachable: yes //The current gateway is
reachable.
VIP 192.168.199.151 reachable:  yes //The VIP is reachable.
Peer 192.168.196.125 reachable: yes //MN2 is reachable.
Keepalived status:             active //The Keepalived service is
in the running state.
```

```

ZStack HA Monitor:          active //The HA monitoring service
is in the running state.
MySQL status:               mysqld is alive //The database is
running properly.
MN status: Running [PID:6500] //The MN is running properly.
UI status: Running [PID:9785] http://192.168.195.200:5000 //The UI
is running properly.

Slave Status:
-----
        Slave_IO_Running: Yes //Slave I/O is running properly.
        Slave_SQL_Running: Yes //Slave SQL is running properly.
        Last_Error:
        Seconds_Behind_Master: 0
        Last_IO_Error:
        Last_SQL_Error:

Warning: Permanently added '192.168.196.125' (ECDSA) to the list of
known hosts.
Status report from 192.168.196.125 //View the status of MN2.
=====
Owns virtual address:       no
Self 192.168.196.125 reachable: yes
Gateway 192.168.0.1 reachable: yes
VIP 192.168.199.151 reachable: yes
Peer 192.168.195.200 reachable: yes
Keepalived status:         active
ZStack HA Monitor:         active
MySQL status:               mysqld is alive

Slave Status:
-----
        Slave_IO_Running: Yes
        Slave_SQL_Running: Yes
        Last_Error:
        Seconds_Behind_Master: 0
        Last_IO_Error:
        Last_SQL_Error:

Note: visit ZStack UI with http://192.168.199.151:5000

```

## 2. Prepare the following necessary packages:

- ZStack Cloud custom ISO:
  - C76: ZStack-Cloud-x86\_64-DVD-4.3.12-c76.iso
  - C74: ZStack-Cloud-x86\_64-DVD-4.3.12-c74.iso
  - Download address: [Click here](#)
- ZStack Cloud installation package:
  - Software: ZStack-Cloud-installer-4.3.12.bin
  - Download address: [Click here](#)
- Upgrade script
  - Software: ZStack-upgrade

— Download address: [Click here](#)



**Note:**

The version of the upgrade script must be consistent with the version of the custom ISO and the bin package.



**Note:**

After you download the software, check the authentication code by using MD5 and verify that the code conforms with the release information.

3. Select a proper upgrade method. You can upgrade from the bin package or from ISO.

If you upgrade the MN from the bin package, follow these steps:

1. Import the new version of ZStack Cloud custom ISO to each MN and run the following command on each MN to upgrade the local repo to the latest version:

```
[root@localhost ~]# bash zstack-upgrade -r ZStack-Cloud-installer-4.3.12.bin
```

2. Run the following command on either MN. Then, both MNs will be upgraded.

```
[root@localhost ~]# zsha2 upgrade-mn -peerpass password ZStack-Cloud-installer-4.3.12.bin
```

If you choose to upgrade the MN from ISO, follow this step:

1. Run the following command on either MN. Then, both MNs will be upgraded.

```
[root@localhost ~]# zsha2 upgrade-mn -peerpass password ZStack-Cloud-x86_64-DVD-4.3.12-c76.iso
```



**Note:**

In the preceding command, the `-peerpass` parameter is optional. You can use this parameter to set the SSH login password for the peer MN.

## 1.4 Other Operations

### 1.4.1 Monitoring Alarm

In a dual-MN HA scenario, if the active MN is disconnected, the administrator can create an event alarm in Monitoring Alarm, add the corresponding alarm metric, and specify an endpoint. Then, the system will send alarm messages via email, DingTalk, HTTP application, short message service, or Microsoft Teams.

**Figure 1-9: Active MN Is Disconnected**

If the backup MN is disconnected, the administrator can receive relevant notifications in Message Center.

**Figure 1-10: Backup MN Is Disconnected**

Message Content	Trigger Action	Emergency Level	Message Type	Alarm Times	Last Alarm Time	Mute for	Confirmer	Conf	Actions
Host Disconnected	Host Disconnected	Emergent	Event Alarm	3	2021-07-30 10:46:42	None	None	None	...
Management Node Connected	Management Node Connected	Info	Event Alarm	7	2021-07-30 10:46:06	None	None	None	...
Management Node Connected	Management Node Connected	Info	Event Alarm	4	2021-07-30 10:45:44	None	None	None	...
Management Node Disconnected	Management Node Disconnected	Emergent	Event Alarm	4	2021-07-30 10:26:59	None	None	None	...

## 1.4.2 Log Output

In a dual-MN HA scenario, the administrator can collect logs related to the **zsha2** service by running the following commands:

```
[root@localhost ~]# zsha2 collect-log
Collecting logs ...
Collected log: zsha2-log-2018-09-17T154358+0800.tgz

# Decompress the log package.
[root@localhost ~]# tar zxvf zsha2-log-2021-01-17T154358+0800.tgz
tmp/zsha2-log588815976/
tmp/zsha2-log588815976/zsha2-status.log
tmp/zsha2-log588815976/zstack-ha.log
tmp/zsha2-log588815976/keepalived.data
tmp/zsha2-log588815976/zs-vip-192.168.199.151.log
tmp/zsha2-log588815976/keepalived_status.log
```

## 2 HA Test and Recovery

---

### 2.1 Planned O&M

#### 2.1.1 Single-MN Maintenance

##### Active MN Maintenance

In a dual-MN HA scenario, assume that MN1 is the active MN and MN2 is the standby MN.

If the administrator needs to temporarily shut down MN1 for maintenance, follow these steps:

1. Switch MN1 to the standby MN.

On MN1, run the `zsha2 demote` command. Then, MN1 is switched to the standby MN online, and MN2 becomes the active MN after obtaining a VIP.

2. Stop MN1.

- If MN1 is not added to ZStack Cloud as a compute node, follow these steps:
  1. Run the `zsha2 stop-node` command on MN1 to stop the **zsha2** service.
  2. Shut down MN1.
  3. Maintain MN1 after shutdown.
- If MN1 is reused as a compute node and added to ZStack Cloud, follow these steps:
  1. Run the `zsha2 stop-node` command on MN1 to stop the **zsha2** service.
  2. Put MN1 into maintenance mode.
  3. Shut down MN1.
  4. Maintain MN1 after shutdown.

3. Start MN1.

- a. After powering on MN1, start the server manually or through IPMI.
- b. Wait for MN1 to start and successfully boot the operating system.
- c. On MN1, run the `zsha2 start-node` command to start the **zsha2** service.
- d. On MN1, run the `zsha2 status` command to check whether the **zsha2** service is running properly.
- e. On MN1, run the `zstack-ctl status` command to check whether the MN service and the UI are running properly.

## Standby MN Maintenance

In a dual-MN HA scenario, assume that MN1 is the active MN and MN2 is the standby MN.

If the administrator needs to temporarily shut down MN2 for maintenance, follow these steps:

### 1. Stop MN2.

- If MN2 is not added to ZStack Cloud as a compute node, follow these steps:
  1. On MN2, run the `zsha2 stop-node` command to stop the **zsha2** service.
  2. Shut down MN2.
  3. Maintain MN2 after shutdown.
- If MN2 is reused as a compute node and added to ZStack Cloud, follow these steps:
  1. On MN2, run the `zsha2 stop-node` command to stop the **zsha2** service.
  2. Put MN2 into the maintenance mode.
  3. Shut down MN2.
  4. Maintain MN2 after shutdown.

### 2. Start MN2.

- a. After powering on MN2, start the server manually or through IPMI.
- b. Wait for MN2 to start and successfully boot the operating system.
- c. On MN2, run the `zsha2 start-node` command to start services related to **zsha2**.
- d. On MN2, run the `zsha2 status` command to check whether the **zsha2** service is running properly.
- e. On MN2, run the `zstack-ctl status` command to check whether the MN service and the UI are running properly.

## 2.1.2 Dual-MN Maintenance

In a dual-MN HA scenario, assume that MN1 is the active MN and MN2 is the standby MN.

If the administrator needs to temporarily shut down these two MNs for maintenance, follow these steps:

1. On each MN, run the `zsha2 stop-node` command to stop the **zsha2** service.
2. Shut down each MN.
3. Maintain each MN after shutdown.
4. After powering on each MN, start the server manually or through IPMI.
5. Wait for each MN to start and successfully boot the operating system.

6. On each MN, run the `zsha2 start-node` command to start the **zsha2** service.
7. On each MN, run the `zsha2 status` command to check whether the **zsha2** service is running properly.
8. On each MN, run the `zstack-ctl status` command to check whether the MN services and the UI are running properly.

## 2.2 MN Troubleshooting

### 2.2.1 Single-MN Troubleshooting

In a dual-MN HA scenario, if one MN fails, repair and recover the node by following these steps:

1. On the failed MN, run the `zsha2 stop-node` command to stop the **zsha2** service.
2. Try to recover the failed node. If the recovery fails, use ZStack Cloud installation package of the same version to repair the node or install a new node.
3. If you need to install a new node, follow these steps:
  - a. Provision a backup server to make the hardware specifications more similar to those of the failed node.
  - b. Install the basic operating system. After the installation is completed, configure the root password and network information, which must be consistent with that of the failed node. For more information, see [Installation and Deployment](#).
  - c. Install an HA suite for the new node. For more information, see [Installation and Deployment](#).
  - d. On the new node, run the `zsha2 status` command to check whether the **zsha2** service is running properly.
  - e. In the new node, run the `zstack-ctl status` command to check whether the MN service and the UI are running properly.

### 2.2.2 Dual-MN Troubleshooting

In a dual-MN HA scenario, if both MNs fail, repair and recover the nodes by following these steps:

1. Try to recover the failed nodes. If the recovery fails, use ZStack Cloud of the same version to repair the nodes or install two new nodes.
2. If you need to install two new nodes, follow these steps:
  - a. Provision two backup servers to make the hardware specifications more similar to those of the failed nodes.

- b. Install the basic operating system on each backup server. After the installation is completed, configure the root password and network information, which must be consistent with that of the failed nodes. For more information, see [Installation and Deployment](#).
- c. Select a MN that is running properly, SSH to the MN via its IP address, and run the following command to recover the database on this MN:

```
[root@localhost ~]# zstack-ctl restore_mysql -f /var/lib/zstack/
mysql-backup/xxx.gz --mysql-root-password MYSQL_PASSWORD
```

- In the preceding command, `/var/lib/zstack/mysql-backup/xxx.gz` is the file path and name of the backup database.
  - `MYSQL_PASSWORD` is the password of the database. Default password:  
`zstack.mysql.password`.
- d. On this node, run the `install_ha` command to reinstall an HA suite. For more information, see [Use the Command Line](#).
  - e. On the new node, run the `zsha2 status` command to check whether the `zsha2` service is running properly.
  - f. In the new node, run the `zstack-ctl status` command to check whether the MN service and the UI are running properly.

## 2.2.3 MN Database Backup and Recovery

### Database Backup

In a multi-MN HA scenario, you can back up a database by using the following method:

- Select a MN that is running properly, SSH to the MN via its IP address, and run the `zstack-ctl dump_mysql` command to manually back up the database.



#### Note:

- VIPs are used to log in to the UI. Do not use VIPs to log in to a MN through SSH.
- After the database is backed up, the database is saved to the `/var/lib/zstack/mysql-backup/` directory as a `.gz` file. Example name: `zstack-backup-db-2019-06-18_00-30-04.gz`.

### Database Recovery

In a multi-MN HA scenario, you can recover a database by following these steps:



1. Select a MN that is running properly, SSH to the MN via its IP address, and run the following command to recover the database on this MN:

```
[root@localhost ~]# zstack-ctl restore_mysql -f /var/lib/zstack/  
mysql-backup/xxx.gz --mysql-root-password MYSQL_PASSWORD
```

- In the preceding command, */var/lib/zstack/mysql-backup/xxx.gz* is the file path and name of the backup database.
  - *MYSQL\_PASSWORD* is the password of the database. Default password:  
zstack.mysql.password.
2. On this node, run the `install_ha` command to reinstall an HA suite. For more information, see [Use the Command Line](#).

## 3 CLI Guidance

**zsha2** has many sub-commands. This section will elaborate on what each sub-command is and how to use them.

### 3.1 Introduction

**zsha2**, designed by ZStack Cloud for multi-MN HA scenarios, is a command that helps you quickly complete various operations in these scenarios.

### 3.2 -h

#### Description

Displays the help information, which allows you to view all sub-commands of **zsha2**.

#### Usage

```
[root@localhost ~]# zsha2 -h
usage:
  zsha2 [ global options ] command [ command options ]

Global options:
  -h,--help          Display this message

Commands:
  install-ha          install two-node HA environment
  stop-node           stop zstack service in HA environment
  start-node          start zstack service in HA environment
  upgrade-mn          upgrade the MN in HA environment
  upgrade-ha          upgrade the HA suites
  demote              demote current node as backup
  status              show HA status
  show-config          show HA configuration
  sample-config       generate sample configuration to setup HA environment
  collect-log         collect HA related log files
  help               show this help message
```

### 3.3 version

#### Description

Displays the version information, including the version No. and commit ID.

#### Usage

```
[root@localhost ~]# zsha2 version
```

```
version 3.1.0.0, commit 2b1b06788e4e1d4b514342db1f381b460f7242e6
```




## 3.4 install-ha

### Description

An installation command. If you installed two MNs, you could run the **zsha2** installation command in a MN to switch to the dual-MN HA mode.

### Usage

Parameter	Description	Example
-nic	The physical device name, which is used to configure a VIP. In production environments, this parameter is usually a network management bridge.	<pre>./zsha2 install-ha -nic br_bond0</pre>
-gateway	The arbitration gateway of the active and standby MNs.	<pre>./zsha2 install-ha -gateway 192.168.0.1</pre>
-slave	Specifies the standby MN.  <div style="background-color: #f0f0f0; padding: 10px; border: 1px solid #ccc;"> <p><b>Note:</b></p> <ul style="list-style-type: none"> <li>During the installation, the database of the standby MN will be overwritten by that of the active MN. Please exercise caution.</li> <li>If the root password contains special shell characters, escape these characters.</li> </ul> </div>	<pre>./zsha2 install-ha -slave "root:password@192.168.196.125"</pre>
-vip	Specifies the VIP for Keepalived communication.	<pre>./zsha2 install-ha -vip 192.168.199.151</pre>
-myip	Optional. Specifies the local IP address.	<pre>./zsha2 install-ha -myip 192.168.195.200</pre>
-db-root-pw	The database root password of the active and standby MNs. Make sure that these two MNs share the same database root password.	<pre>./zsha2 install-ha -db-root-pw zstack.mysql.password</pre>

Parameter	Description	Example
-time-server	<p>Specifies a time synchronization server for unified time synchronization.</p> <div>  <b>Note:</b>            You can specify multiple time servers at a time.         </div>	<ul style="list-style-type: none"> <li><code>./zsha2 install-ha -time-server 192.168.196.125</code></li> <li><code>./zsha2 install-ha -time-server 192.168.196.125,192.168.196.126</code></li> </ul>
-cidr	<p>Optional. Specifies an network range, which must cover the IP address, VIP, and gateway of the active and standby MNs.</p> <div>  <b>Note:</b>            If not specified, the system will calculate a minimum network range, which might fail to meet the requirements. We recommend that you specify a network range.         </div>	<code>./zsha2 install-ha -cidr 192.168.0.0/16</code>
-force	<p>Optional. Force runs the <b>zsha2</b> installation command in the active MN if the database of the active and standby MNs cannot automatically synchronized for a long time.</p> <div>  <b>Note:</b>            We recommend that you back up these two databases before you perform the force installation.         </div>	<code>./zsha2 install-ha -force</code>
-repo	Optional. Specifies the YUM repository, which defaults to the local repository.	<code>./zsha2 install-ha -repo zstack-local</code>
-timeout	Optional. The timeout for copying the databases of the active and standby MNs. Default: 600. Unit: second.	<code>./zsha2 install-ha -timeout 600</code>

Parameter	Description	Example
-yes	Optional. Indicates that all settings are allowed.	<code>./zsha2 install-ha -yes</code>
-config	Optional. Initializes and installs an HA suite by using a configuration file.	<code>./zsha2 install-ha -config zs-install.config</code>

The following is an example of how to install an HA suite by specifying the command line:

```
[root@localhost ~]# ./zsha2 install-ha -nic br_bond0 -gateway 192.168.0.1 -slave "root:password@192.168.196.125" \
-vip 192.168.199.151 -myip 192.168.195.200 -db-root-pw zstack.mysql.password -time-server 192.168.196.125 -cidr 192.168.0.0/16 -yes
Master IPv4 address: 192.168.195.200
ZStack version @ 192.168.195.200: 2.6.0
ZStack version @ 192.168.196.125: 2.6.0
Calculated CIDR: 192.168.0.0/16
```

```
Backupping databases on 192.168.196.125 (/var/lib/zstack/mysql-backup/zstack-backup-db-2018-10-09T164934-0800.gz) ...
```

```
Start installation ...
```

```
x checking network interface and gateway ...
✓ Task 1: checking network interface and gateway ... completed.
x prepare HA-services ...
✓ Task 2: prepare HA-services ... completed.
+ setting up DB config before replication ...
✓ Task 3: setting up DB config before replication ... completed.
x creating DB user for replication ...
✓ Task 4: creating DB user for replication ... completed.
+ update iptables rules ...
✓ Task 5: update iptables rules ... completed.
+ starting the initial replication ...
***** 1. row *****
File: mysql-bin.000002
Position: 1844
Binlog_Do_DB:
Binlog_Ignore_DB:

+ starting the initial replication ...
Local database backed up to /var/lib/zstack/mysql-backup/zstack-backup-db-2018-10-09T164934-0800.gz

✓ Task 6: starting the initial replication ... completed.
x wait peer slave sync status ...

Slave_IO_Running: Yes
Slave_SQL_Running: Yes
Last_IO_Error:
Last_SQL_Error:
Last_Error:
Last_Errno: 0

✓ Task 7: wait peer slave sync status ... completed.
```

```

+ wait local DB sync status ...
***** 1. row *****
      File: mysql-bin.000002
      Position: 245
      Binlog_Do_DB:
      Binlog_Ignore_DB:

x wait local DB sync status ...

  Slave_IO_Running: Yes
  Slave_SQL_Running: Yes
    Last_IO_Error:
    Last_SQL_Error:
      Last_Error:
      Last_Errno: 0

✓ Task 8: wait local DB sync status ... completed.
+ setting up keepalived ...
✓ Task 9: setting up keepalived ... completed.
x check slave virtual IP settings ...
✓ Task 10: check slave virtual IP settings ... completed.
x configuring ZStack servers ...
✓ Task 11: configuring ZStack servers ... completed.
x installing HA scripts ...
✓ Task 12: installing HA scripts ... completed.
x starting ZStack HA service ...
✓ Task 13: starting ZStack HA service ... completed.
x waiting management node up and running ...
✓ Task 14: waiting management node up and running ... completed.

OK, installation completed.

Hints:
- Stop server with:    zsha2 stop-node,
- Start server with:  zsha2 start-node,
- Get HA status with: zsha2 status -peer 192.168.196.125

Please also setup SSH pubkey-login between 192.168.195.200 and 192.168.196.125

```

The following is an example of how to install an HA suite by writing a configuration file:

```

[root@localhost ~]# ./zsha2 install-ha -config zs-install.config
Master IPv4 address: 192.168.195.200
ZStack version @ 192.168.195.200: 2.6.0
ZStack version @ 192.168.196.125: 2.6.0
Calculated CIDR: 192.168.0.0/16

Backuping databases on 192.168.196.125 (/var/lib/zstack/mysql-backup/
zstack-backup-db-2018-10-09T164934-0800.gz) ...

Start installation ...

x checking network interface and gateway ...
✓ Task 1: checking network interface and gateway ... completed.
x prepare HA-services ...
✓ Task 2: prepare HA-services ... completed.
+ setting up DB config before replication ...
✓ Task 3: setting up DB config before replication ... completed.

```

```

x creating DB user for replication ...
✓ Task 4: creating DB user for replication ... completed.
+ update iptables rules ...
✓ Task 5: update iptables rules ... completed.
+ starting the initial replication ...
***** 1. row *****
      File: mysql-bin.000002
      Position: 1844
      Binlog_Do_DB:
      Binlog_Ignore_DB:

+ starting the initial replication ...
Local database backed up to /var/lib/zstack/mysql-backup/zstack-backup-
db-2018-10-09T164934-0800.gz

✓ Task 6: starting the initial replication ... completed.
x wait peer slave sync status ...

      Slave_IO_Running: Yes
      Slave_SQL_Running: Yes
      Last_IO_Error:
      Last_SQL_Error:
      Last_Error:
      Last_Errno: 0

✓ Task 7: wait peer slave sync status ... completed.
+ wait local DB sync status ...
***** 1. row *****
      File: mysql-bin.000002
      Position: 245
      Binlog_Do_DB:
      Binlog_Ignore_DB:

x wait local DB sync status ...

      Slave_IO_Running: Yes
      Slave_SQL_Running: Yes
      Last_IO_Error:
      Last_SQL_Error:
      Last_Error:
      Last_Errno: 0

✓ Task 8: wait local DB sync status ... completed.
+ setting up keepalived ...
✓ Task 9: setting up keepalived ... completed.
x check slave virtual IP settings ...
✓ Task 10: check slave virtual IP settings ... completed.
x configuring ZStack servers ...
✓ Task 11: configuring ZStack servers ... completed.
x installing HA scripts ...
✓ Task 12: installing HA scripts ... completed.
x starting ZStack HA service ...
✓ Task 13: starting ZStack HA service ... completed.
x waiting management node up and running ...
✓ Task 14: waiting management node up and running ... completed.

OK, installation completed.

Hints:
- Stop server with:  zsha2 stop-node,
- Start server with: zsha2 start-node,
- Get HA status with: zsha2 status -peer 192.168.196.125

```

```
Please also setup SSH pubkey-login between 192.168.195.200 and 192.168.196.125
```

## 3.5 stop-node

### Description

Stops a MN and its **zsha2** service in a dual-MN HA scenario.

### Usage

```
[root@localhost ~]# zsha2 stop-node
stopping zstack-ha service ...
stopping zstack management node ...
stopping keepalived ...
```

## 3.6 start-node

### Description

Starts a MN and its **zsha2** service in a dual-MN HA scenario.

### Usage

```
[root@localhost ~]# zsha2 start-node
starting keepalived ...
starting zstack-ha service ...
starting zstack management node ...
```

## 3.7 upgrade-mn

### Description

Upgrades only MNs in a dual-MN HA scenario.

### Usage

Parameter	Description	Example
-force	Optional. Force upgrades MNs	<code>zsha2 upgrade-mn -force ZStack-Cloud-installer-4.3.12.bin</code>
-peerpass	Optional. Enters the SSH login password of the peer MN.	<code>zsha2 upgrade-mn -peerpass password ZStack-Cloud-installer-4.3.12.bin</code>
-yes	Optional. Indicates that all settings are allowed.	<code>zsha2 upgrade-mn -yes</code>



If you upgrade the MN from the bin package, follow these steps:

1. Import the new version of ZStack Cloud custom ISO to each MN and run the following command on each MN to upgrade the local repo to the latest version:

```
[root@localhost ~]# bash zstack-upgrade -r ZStack-Cloud-installer-4.3.12.bin
```

2. Run the following command on either MN. Then, both MNs will be upgraded.

```
[root@localhost ~]# zsha2 upgrade-mn -peerpass password ZStack-Cloud-installer-4.3.12.bin
```

If you choose to upgrade the MN from ISO, follow this step:

1. Run the following command on either MN. Then, both MNs will be upgraded.

```
[root@localhost ~]# zsha2 upgrade-mn -peerpass password ZStack-Cloud-x86_64-DVD-4.3.12-c76.iso
```



**Note:**

In the preceding command, the `-peerpass` parameter is optional. You can use this parameter to set the SSH login password for the peer MN.



**Note:**

In the preceding command, the `-peerpass` parameter is optional. You can use this parameter to set the SSH login password for the peer MN.

## 3.8 upgrade-ha

### Description

Upgrades the **zsha2** service on the current node in a dual-MN scenario.

### Usage

```
[root@localhost ~]# ./zsha2 upgrade-ha

Start upgrading ...

+ Stopping HA-services ...
✓ Task 1: Stopping HA-services ... completed.
+ Upgrading HA suites ...
✓ Task 2: Upgrading HA suites ... completed.
x starting ZStack HA service ...
✓ Task 3: starting ZStack HA service ... completed.

OK, upgrade HA completed.
```

**Hints:**

- Stop server with: `zsha2 stop-node`,
- Start server with: `zsha2 start-node`,
- Get HA status with: `zsha2 status -peer 192.168.196.125`

## 3.9 demote

### Description

Switches the active MN to the standby MN online in a dual-MN HA scenario.

### Usage

```
[root@localhost ~]# zsha2 demote
```

## 3.10 status

### Description

In a dual-MN HA scenario, displays the status of the current MN, including whether a VIP is obtained, whether the MN is reachable, whether the gateway is reachable, whether the VIP is reachable, whether the peer MN is reachable, the status of the Keepalived service, the status of the HA monitoring service, the database status, the MN status, the UI status, the Slave status, and the status of the peer MN.

### Usage

```
[root@localhost ~]# zsha2 status
Status report from 192.168.195.200
=====
Owns virtual address:          yes
Self 192.168.195.200 reachable: yes
Gateway 192.168.0.1 reachable: yes
VIP 192.168.199.151 reachable: yes
Peer 192.168.196.125 reachable: yes
Keepalived status:            active
ZStack HA Monitor:            active
MySQL status:                  mysqld is alive
MN status: Running [PID:6500]
UI status: Running [PID:9785] http://192.168.195.200:5000

Slave Status:
-----
Slave_IO_Running: Yes
Slave_SQL_Running: Yes
Last_Error:
Seconds_Behind_Master: 0
Last_IO_Error:
Last_SQL_Error:

Warning: Permanently added '192.168.196.125' (ECDSA) to the list of
known hosts.
Status report from 192.168.196.125
=====
```

```
Owns virtual address:      no
Self 192.168.196.125 reachable:  yes
Gateway 192.168.0.1 reachable:  yes
VIP 192.168.199.151 reachable:  yes
Peer 192.168.195.200 reachable:  yes
Keepalived status:        active
ZStack HA Monitor:        active
MySQL status:             mysqld is alive
```

Slave Status:

```
-----
                Slave_IO_Running: Yes
                Slave_SQL_Running: Yes
                Last_Error:
Seconds_Behind_Master: 0
                Last_IO_Error:
                Last_SQL_Error:
```

Note: visit ZStack UI with <http://192.168.199.151:5000>

## 3.11 show-config

### Description

In a dual-MN HA scenario, displays the configuration information about the current MN.

### Usage

```
[root@localhost ~]# zsha2 show-config
{
  "nodeip": "192.168.195.200",
  "peerip": "192.168.196.125",
  "dbvip": "192.168.199.151",
  "nic": "br_bond0",
  "gw": "192.168.0.1",
  "dbnetwork": "192.168.0.0/16",
  "repo": "zstack-local",
  "version": 0
}
```

## 3.12 sample-config

### Description

In a dual-MN HA scenario, generates sample configurations to set up an HA environment quickly.

### Usage

```
[root@localhost ~]# zsha2 sample-config
{
  "gateway": "172.20.0.1",
  "virtualIp": "172.20.0.2",
  "myIp": "172.20.0.3",
  "peerIp": "172.20.0.4",
  "peerSshUser": "root",
  "peerSshPass": "somepass",
  "peerSshPort": 22,
```

```
"dbRootPass": "zstack.password",  
"interface": "br_eth0",  
"timeServer": "172.20.0.3"  
}
```

## 3.13 collect-log

### Description

Collects logs related to the **zsha2** service in a dual-MN HA scenario.

### Usage

```
[root@localhost ~]# zsha2 collect-log  
Collecting logs ...  
Collected log: zsha2-log-2018-09-17T154358+0800.tgz  
  
# Decompress the log package.  
[root@localhost ~]# tar zxvf zsha2-log-2018-09-17T154358+0800.tgz  
tmp/zsha2-log588815976/  
tmp/zsha2-log588815976/zsha2-status.log  
tmp/zsha2-log588815976/zstack-ha.log  
tmp/zsha2-log588815976/keepalived.data  
tmp/zsha2-log588815976/zs-vip-192.168.199.151.log  
tmp/zsha2-log588815976/keepalived_status.log
```

# Glossary

---

## VM Instance

A VM instance is a virtual machine instance running on a host. A VM instance has its own IP address and can access public networks and run application services.

## Volume

A volume provides storage space for a VM instance. Volumes are categorized into root volumes and data volumes.

## Root Volume

A root volume provides support for the system operations of a VM instance.

## Data Volume

A data volume provides extended storage space for a VM instance.

## Image

An image is a template file used to create a VM instance or volume. Images are categorized into system images and volume images.

## Instance Offering

An instance offering defines the number of vCPU cores, memory size, network bandwidth, and other configuration settings of VM instances.

## Disk Offering

A disk offering defines the capacity and other configuration settings of volumes.

## GPU Specification

A GPU specification defines the frame per second (FPS), video memory, resolution, and other configuration settings of a physical or virtual GPU. GPU specifications are categorized into physical GPU specifications and virtual GPU specifications.

## Auto-Scaling Group

An auto-scaling group is a group of VM instances that are used for the same scenarios. An auto-scaling group can automatically scale out or in based on application workloads or health status of VM instances in the group.

## Snapshot

A snapshot is a point-in-time capture of data status in a volume.

## Affinity Group

An affinity group is an orchestration policy for IaaS resources to ensure the high performance and high availability of businesses...

## Zone

A zone is a logical group of resources such as clusters, L2 networks, and primary storages. Zone is the largest resource scope defined in the Cloud.

## Cluster

A cluster is a logical group of hosts (compute nodes).

## Host

A host provides compute, network, and storage resources for VM instances.

## Primary Storage

A primary storage is one or more servers that store volume files of VM instances. These files include root volume snapshots, data volume snapshots, image caches, root volumes, and data volumes.

## Backup Storage

A backup storage is one or more servers that store VM image templates, including ISO image files .

## iSCSI Storage

iSCSI storage is a SAN storage that uses the iSCSI protocol for data transmission. You can add an iSCSI SAN block as a Shared Block primary storage or pass through the block to a VM instance.

## FC Storage

FC storage is an SAN storage that uses the FC technology for data transmission. You can add an FC SAN block as a Shared Block primary storage or pass through the block to a VM instance.

## L2 Network

An L2 network is a layer 2 broadcast domain used for layer 2 isolation. Generally, L2 networks are identified by names of devices on the physical network.

## VXLAN Pool

A VXLAN pool is a collection of VXLAN networks established based on VXLAN Tunnel Endpoints (VTEPs). The VNI of each VXLAN network in a VXLAN pool must be unique.

## L3 Network

An L3 network includes IP ranges, gateway, DNS, and other network configurations that are used by VM instances.

## Public Network

Generally, a public network is a logical network that is connected to the Internet. However, in an environment that has no access to the Internet, you can also create a public network.

## Flat Network

A flat network is connected to the network where the host is located and has direct access to the Internet. VM instances in a flat network can access public networks by using elastic IP addresses.

## VPC Network

A VPC network is a private network where VM instances can be created. A VM instance in a VPC network can access the Internet through a VPC vRouter.

## Management Network

A management network is used to manage physical resources in the Cloud. For example, you can create a management network to manage access to hosts, primary storages, backup storages, and VPC vRouters.

## Flow Network

A flow network is a dedicated network for port mirror transmission. You can use a flow network to transmit the mirrors of data packets of NIC ports to the target ports.

## VPC vRouter

A VPC vRouter is a dedicated VM instance that provides multiple network services.

## VPC vRouter HA Group

A VPC vRouter HA group consists of two VPC vRouters. Either VPC vRouter can be a primary or secondary VPC vRouter for the group. If the primary VPC vRouter does not work as expected, the VPC vRouter becomes the secondary VPC vRouter in the group to ensure high availability of business.

## vRouter Image

A vRouter image encapsulates network services and can be used to create VPC vRouters and load balancers. vRouter images can be categorized into VPC vRouter images and load balancer (LB) images.

## Dedicated-Performance LB Image

A dedicated-performance load balancer (LB) image encapsulates dedicated-performance load-balancing services and can be used to create load balancer instances. However, a dedicated-performance load balancer image cannot be used to create VM instances.

## vRouter Offering

A vRouter offering defines the number of vCPU cores, memory size, image, management network, and public network configuration settings of VPC vRouters. You can use a vRouter offering to create VPC vRouters that can provide network services for public networks and VPC networks.

## LB Instance Offering

A load balancer (LB) instance offering defines the CPU, memory, image, and management network configuration settings used to create LB instances. LB instances provide load balancing services for the public network, flat network, and VPC network.

## SDN Controller

An SDN controller is used to control network devices such as switches. You can add an external SDN controller to the Cloud and use the controller to control external switches and other network devices.



## Security Group

A security group provides security control services for VM instances on the L3 network. It filters the ingress or egress TCP, UDP, and ICMP packets of specified VM instances in specified networks based on the specified security rules.

## VIP

In bridged network environments, a virtual IP address (VIP) provides network services such as serving as an elastic IP address (EIP), port forwarding, load balancing, IPsec tunneling. When a VIP provides the preceding network services, packets are sent to the VIP and then routed to the destination network where VM instances are located.

## EIP

An elastic IP address (EIP) functions based on the NAT technology. IP addresses in a private network are translated into an EIP that is in another network. This way, private networks can be accessed from other networks by using EIPs.

## Port Forwarding

Port forwarding functions based on the layer-3 forwarding service of VPC vRouters. This service forwards traffic flows of the specified IP addresses and ports in a public network to specified ports of VM instances by using the specified protocol. If your public IP addresses are insufficient, you can configure port forwarding for multiple VM instances by using one public IP address and port.

## Load Balancer

A load balancer distributes traffic flows of a virtual IP address to backend servers. It automatically inspects the availability of backend servers and isolates unavailable servers during traffic distribution. This way, the load balancer improves the availability and service capability of your business.

## Listener

A listener monitors the frontend requests of a load balancer and distributes the requests to a backend server based on the specified policy. In addition, the listener performs health checks on backend servers.

## Forwarding Rule

A forwarding rule forwards the requests from different domain names or URLs to different backend server groups.

## Backend Server Group

A backend server group is a group of backend servers that handles requests distributed by load balancers. It is the basic unit for traffic distribution by load balancer instances.

## Backend Server

A backend server handles requests distributed by a load balancer. You can add a VM instance on the Cloud or a server on a third-party cloud as a backend server.

## Frontend Network

A frontend network is a type of network that is associated with a load balancer. Requests from the network are distributed by the load balancer to backend servers based on a specified policy.

## Backend Network

A backend network is a type of network that is associated with a load balancer. Requests from frontend networks are distributed by the load balancer to servers in the backend network.

## Load Balancer Instance

A load balancer instance is a custom VM instance used to provide load balancing services.

## Certificate

If you select HTTPS for a listener, associate it with a certificate to make the listener take effect. You can upload either a certificate or certificate chain.

## Firewall

A firewall is an access control policy that monitors ingress and egress traffic of VPC vRouters and decides whether to allow or block specific traffic based on a defined set of security rules.

## IPsec Tunnel

An IPsec tunnel encrypts and verifies IP packets that transmit over a virtual private network (VPN ) from one site to another.

## OSPF Area

An OSPF area is split from an autonomous system based on the OSPF protocol. This splitting simplifies the management of vRouters.

## NetFlow

An NetFlow monitors the ingress and egress traffic of the NICs of VPC vRouters. The supported versions of data flows are V5 and V9.

## Port Mirroring

Port mirroring mirrors the traffic data of VM NICs and sends the traffic data to the target ports. This allows for the analysis of data packets of ports and simplifies the monitoring and management of data traffic and makes it easier to locate network errors and exceptions.

## Route Table

A route table contains information about various routes that you configure. Route entries in a route table must include the destination network, next hop, and route priority.

## CloudFormation

CloudFormation is a service that simplifies the management of cloud resources and automates deployment and O&S. You can create a stack template to configure cloud resources and their dependencies. This way, resources can be automatically configured and deployed in batches. CloudFormation provides easy management of the lifecycle of cloud resources and integrates automatic O&S into API and SDK.

## Resource Stack

A resource stack is a stack of resources that are configured by using a stack template. The resources in the stack have dependencies with each other. You can manage resources in the stack by managing the resource stack.

## Stack Template

A stack template is a UTF8-encoded file based on which you can create resource stacks. The stack template defines the resources that you want, the dependencies between the resources , and the configuration settings of the resources. When you use a stack template to create a resource stack, CloudFormation parses the template and the resources are automatically created and configured.

## Sample Template

A sample template is a commonly used resource stack. You can use a sample template provide by the Cloud to create resource stacks.

## Designer

A designer is a CloudFormation tool that allows you to orchestrate cloud resources. You can drag and drop resources on a canvas and use lines to establish dependencies between the resources.

## Baremetal Cluster

A baremetal cluster consists of baremetal chassis. You can manage baremetal chassis by managing a baremetal cluster where the chassis reside.

## Deployment Server

A deployment server is a server that provides PXE service and console proxy service for baremetal chassis.

## Baremetal Chassis

A baremetal chassis is used to create a baremetal instance and is identified based on the BMC interface and IPMI configuration setting.

## Preconfigured Template

A preconfigured template is used to create a preconfigured file that allows for unattended batch installation of an operating system for baremetal instances.

## Baremetal Instance

A baremetal instance is an instantiated baremetal chassis.

## Elastic Baremetal Management

Elastic Baremetal Management provides dedicated physical servers for your applications to ensure high performance and stability. In addition, this feature allows elastic scaling. You can apply for and scale resources based on your needs.

## Provision Network

A provision network is a dedicated network for PXE boot and image downloads while creating elastic baremetal instances.

## Elastic Baremetal Cluster

An elastic baremetal cluster consists of elastic baremetal instances. You can manage elastic baremetal instances by managing an elastic baremetal cluster where the instances reside.

## Gateway Node

A gateway node is a node where the ingress and egress traffic of the Cloud and elastic baremetal instances is forwarded.

## Baremetal Node

A baremetal node is used to create a baremetal instance and is identified based on the BMC interface and IPMI configuration setting.

## Elastic Baremetal Instance

An elastic baremetal instance has the same performance as physical servers and allows elastic scaling. You can apply for and scale resources based on your needs.

## Elastic Baremetal Offering

An elastic baremetal offering defines the number of vCPU cores, memory size, CPU architecture, CPU model, and other configuration settings of elastic baremetal instances.

## vCenter

The Cloud allows you to take over vCenter and manage resources on the vCenter.

## VM Instance

A VM instance is an ESXi virtual machine instance running on a host. A VM instance has its own IP address to access public networks and can run application services.

## Network

A vCenter network defines the network settings of VM instances on vCenter, such as IP range, gateway, DNS, and network services.

## Volume

A volume provides storage space for a VM instance on vCenter. A volume attached to a VM instance can be used as a root volume or data volume. A root volume provides support for the system operations of a VM instance. A data volume provides extended storage space for a VM instance.

## Image

An image is a template file used to create a VM instance or volume on vCenter. Images are categorized into system images and volume images.

## Event Message

Event Message displays event alarm messages of vCenter that is took over by the Cloud. This feature allows you to locate errors and exceptions efficiently.

## Network Topology

A network topology visualizes the network architecture of the Cloud. It allows for efficient planning , management, and improvement of network architecture. Network topologies can be categorized into global topologies and custom topologies.

## Performance Analysis

Performance Analysis displays the performance metrics of key resources under monitoring in the Cloud. Cloud resources can be externally or internally monitored. You can use either method to monitor the performance of resources in the Cloud and improve O&S efficiency.

## Capacity Management

Capacity Management visualizes the capacities and usages of key resources in the Cloud. You can use this feature to improve O&S efficiency.

## MN Monitoring

MN monitoring allows you to view the health status of each management node when you use multiple management nodes to achieve high availability.

## Alarm

An alarm is used to monitor the status of time-series data and events and respond to the status change. Alarms can be categorized into resource alarm, event alarm, and extended alarm.

## One-Click Alarm

A one-click alarm integrates multiple metrics of a resource. You can create one-click alarms for multiple resources to monitor these resources.

## Alarm Template

An alarm template is a template of alarm rules. If you associate an alarm template with a resource group, an alarm is created to monitor the resources in the group.

## Resource Group

A resource group consists of resources grouped based on your business needs. If you associate an alarm template with a resource group, the alarm rules specified by the template take effect on all the resources in the group.

## Message Template

A message template specifies the text template of a resource alarm message or event alarm message sent to an SNS system.

## Message Source

A message source is used to take over extended alarm messages. If you configure alarms for message sources, extended alarm messages can be sent to various endpoints.

## Endpoint

An endpoint is a method that users obtain subscribed messages. Endpoints are categorized into system endpoints, email, DingTalk, HTTP application, short message service, and Microsoft Teams.

## Alarm Message

An alarm message is a message sent the time when an alarm is triggered.

## Operation Log

An operation log is a chronological record of operations on the specified objects and their operation results.

## Audit

Audit monitors and records all activities on the Cloud. You can use this feature to implement operation tracking, cybersecurity classified protection compliance, security analysis, troubleshooting, and automatic O&M.

## Backup Management

Backup management integrates multiple disaster recovery technologies such as incremental backup and full backup that are suitable for multiple business scenarios. You can implement local backup and remote backup based on your business needs.

## Backup Job

You can create a backup job to back up local VM instances, volumes, or databases to a specified storage server on a regular basis.

## Local Backup Data

Local backup data of VM instances, volumes, and databases is stored in the local backup storage.

## Local Backup Storage

A local backup storage is located at the local data center and is used to store local backup data.

## Remote Backup Storage

A remote backup storage is located at a remote data center or a public cloud and is used to store remote backup data.

## Continuous Data Protection (CDP)

Continuous Data Protection (CDP) provides second-level and fine-grained continuous backups for important business systems in VM instances, allowing users to restore VM data to any time state and retrieve files without restoring the system.

## CDP Task

You can create a CDP task to continuously back up your VM data to a specified backup storage to achieve continuous data protection and restoration.

## CDP Data

The backup data generated from continuous data protection on VM instances is stored in local backup storages.

## Scheduled Job

A scheduled job defines that a specific action be implemented at a specified time based on a scheduler.

## Scheduler

A scheduler is used to schedule jobs. It is suitable for business scenarios that last for a long time.

## Tag

A tag is used to mark resources. You can use a tag to search for and aggregate resources.



## Migration Service

The Cloud provides V2V migration service that allows you to migrate VM instances and data from other virtualized platform to the current cloud platform.

## V2V Migration

V2V Migration allows you to migrate VM instances from the VMware or KVM platform to the current cloud platform.

## V2V Conversion Host

A V2V conversion host is a host in the destination cluster that you need to specify during V2V migration to cache VM instances and data when you implement V2V migration. After the VM instances and data are cached in the V2Vconversion host, they are migrated to the destination primary storage.

## User

A user is a natural person that constructs the most basic unit in business management.

## Member Group

A member group is a collection of natural persons or a collection of project members. You can use a member group to grant permissions.

## Role

A role is a collection of permissions that can be granted to users. A user that assumes a role can call API operations based on the permissions specified by the role. Roles are categorized into system roles and custom roles.

## 3rd Party Authentication

The 3rd party authentication feature allows you to integrate third-party authentication systems to the Cloud. Then you can use a third-party account to log in to the Cloud and use the resources in the Cloud. You can add an AD or LDAP server to the Cloud.

## Project

A project is a task that needs to be accomplished by specific personnel at a specified time.

Resources and budgets are also specified for projects. In business management, you can plan resources at the project granularity and allocate an independent resource pool for a project.

## Project Member

A project member is a member in a project who is granted permissions on specific project resources and can use the resources to accomplish tasks. Project members include the project admin, project managers, and normal project members.

## Process Management

Process management is part of ticket management that manages the processes related to the resources of projects. Processes can be categorized into default processes and custom processes .

## My Approvals

In the Cloud, only the administrator and project administrators are granted approval permissions. the administrator and project administrators can approve or reject a ticket. If a ticket is approved, resources are automatically deployed and allocated to the specified project.

## Bills

A bill is the expense of resources totaled at a specified time period. Billing is accurate to the second. Bills can be categorized into project bills, department bills, and account bills.

## Pricing List

A pricing list is a list of unit prices of different resources. The unit price of a resource is set based on the specification and usage time of the resource.

## Console Proxy

Console proxy allows you to log in to a VM instance by using the IP address of a proxy.

## AccessKey Management

An AccessKey pair is a security credential that one party authorizes another party to call API operations and access its resources in the Cloud. AccessKey pairs shall be kept confidential.

## IP Blocklist/Allowlist

An IP blocklist or allowlist identifies and filters IP addresses that access the Cloud. You can create an IP allowlist or blocklist to improve access control of the Cloud.

## Application Center

Application Center allows you to add third-party applications to the Cloud and then access the applications by using the Cloud. It extends the functionality of the Cloud.

## Sub-Account Management

A sub-account is created and managed by the admin. Resources created under a sub-account is managed by the sub-account.

## Theme and Appearance

You can customize the theme and appearance of the Cloud.

## Email Server

If you select Email as the endpoint of an alarm, you need to set an email server. Then alarm messages are sent to the email server.

## Log Server

A log server is used to collect logs of the management node. You can add a log server to the cloud and use the collected logs to locate errors and exceptions. This makes your O&M more efficient.

## Global Setting

Global Setting allows you to configure settings that take effect on the whole platform.

## Scenario Template

Scenario Template provides multiple templates that encapsulate scenario-based global settings . You can apply a template globally with one click based on your business needs. This improves your O&M efficiency.