NVIDIA DGX SuperPOD

Deployment Guide

Featuring NVIDIA DGX A100 Systems

DG-11251-001| February 2023

Document History

DG-11251-001

Version	Date	Authors	Description of Change
0.5	2022-12-22	Alex James, Davinder Singh, Greg Zynda, Mark Troyer, Rangam Addepalli, Robert Sohigian, Robert Strober, Scott Ellis, and Yang Yang	Early access
0.7	2023-01-18	Alex James, Charles Kim, Craig Tierney, and Robert Sohigian	Minor updates
1	2023-02-08	Rangam Addepalli and Robert Sohigian	Base Command Manager 3.23.01

Contents

1 Initial Cluster Setup

This document details how to deploy NVIDIA Base Command™ Manager on NVIDIA DGX SuperPOD™ configurations.

Deployment of a DGX SuperPOD involves pre-setup, deployment, and use of Base Command Manager to provision the Slurm cluster.

Physical installation and network switch configuration should be completed before using this document, along with capturing information about the intended deployment in a site survey. The deployment stage of a DGX SuperPOD consists of using the Base Command Manager to provision and manage the Slurm cluster.

1. Configure the NFS server.

   User home directories (`home/`) and shared data (`cm_shared/`) directories must be shared between head nodes (such as the DGX OS image) must be stored on an NFS filesystem for HA availability. Because DGX SuperPOD does not mandate the nature of the NFS storage, the configuration is outside the scope of this document.

   This DGX SuperPOD deployment uses the NFS export path provided in the site survey: `/var/nfs/general`.

   Following parameters are recommended for the NFS server export file `/etc/exports`.

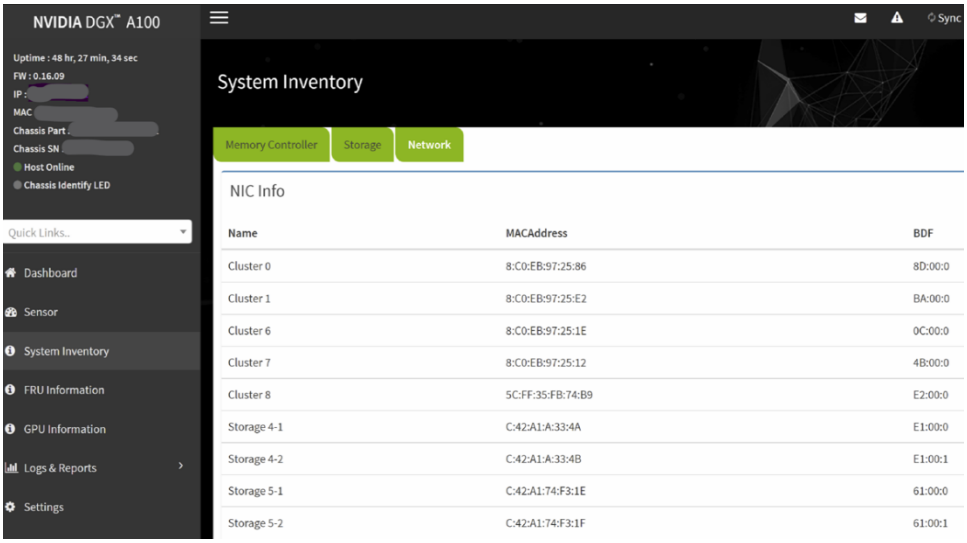
   ```
   /var/nfs/general *(rw,sync,no_root_squash,no_subtree_check)
   ```

2. On the DGX A100 compute nodes, configure the SBIOS so that they PXE boot by default.

   Base Command Manager requires DGX systems to PXE boot.

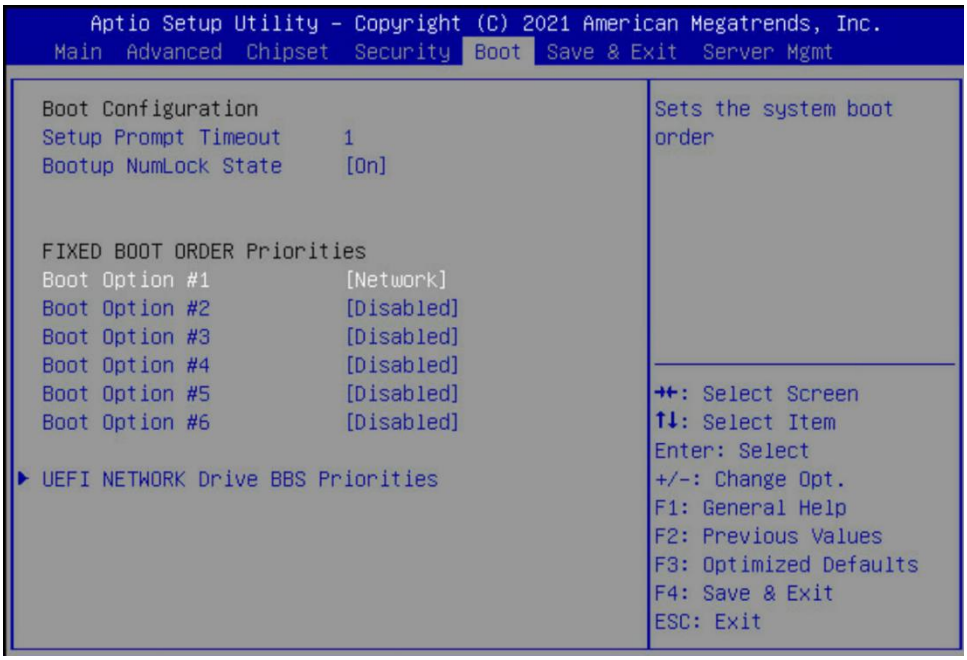
   a. Connect to the BMC web interface of the DGX system.

b. In the `Network` tab of the `System Inventory` window, locate the MAC addresses for the `Storage 4-2` and `Storage 5-2` interfaces.

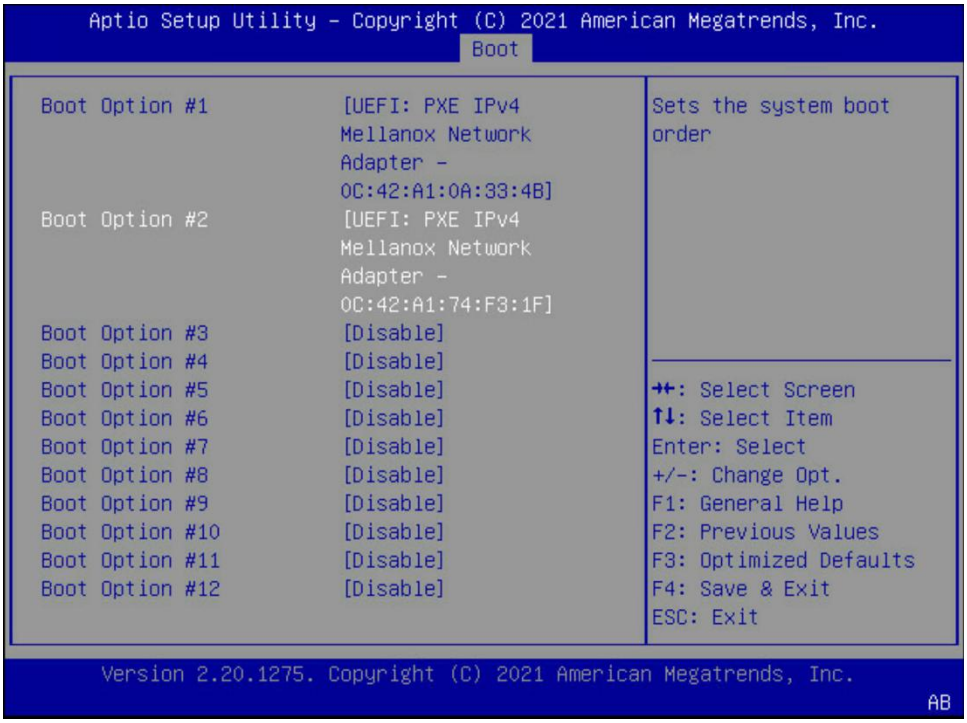


c. Via Remote Control in the Web GUI, enter the DGX A100 system BIOS menu, and configure `Boot Option #1` to be `[NETWORK]`.

Set other Boot devices to `[DISABLED]`.



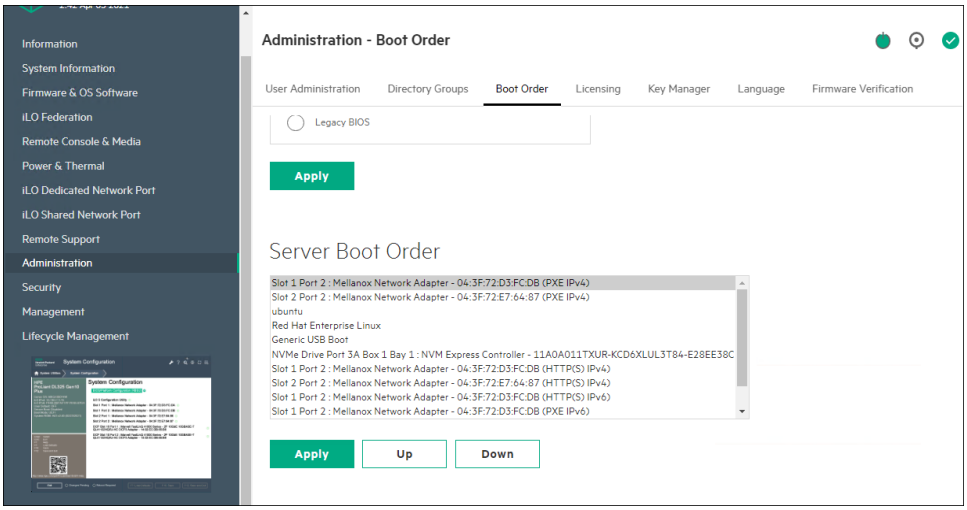
d. Disable PXE boot devices except for `Storage 4-2` and `Storage 5-2`. Set them to use IPv4.



e. Select `Save & Exit` the BIOS.

3. On the failover head node and the cpu nodes, ensure that Network boot is configured as the primary option. Ensure that the Mellanox ports connected on the network on the head and cpu nodes are set to Ethernet mode as well.

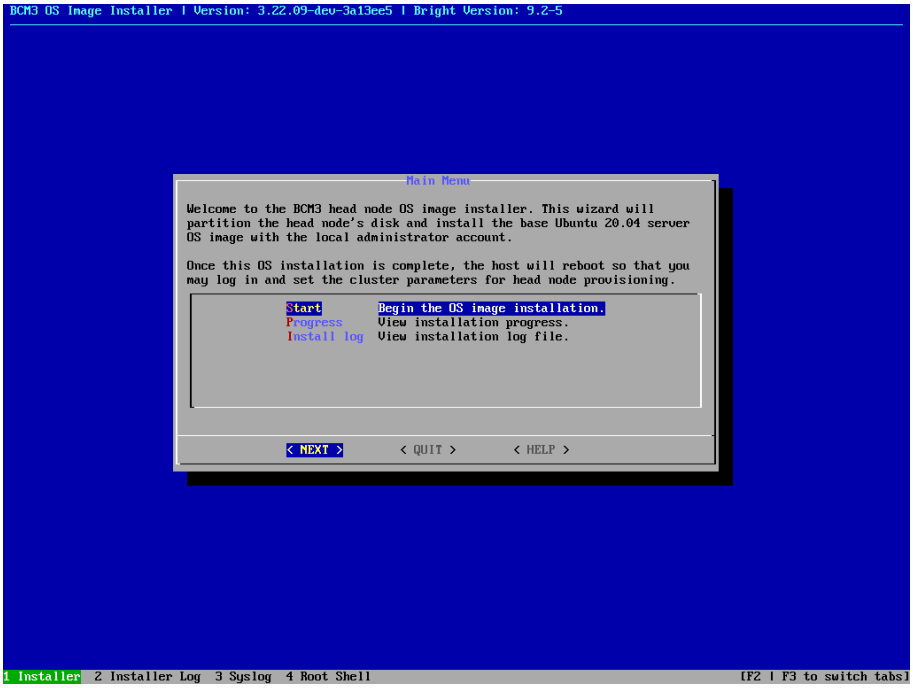
   This is an example of a system that will boot from the network with `Slot 1 Port 2` and `Slot 2 Port 2`.



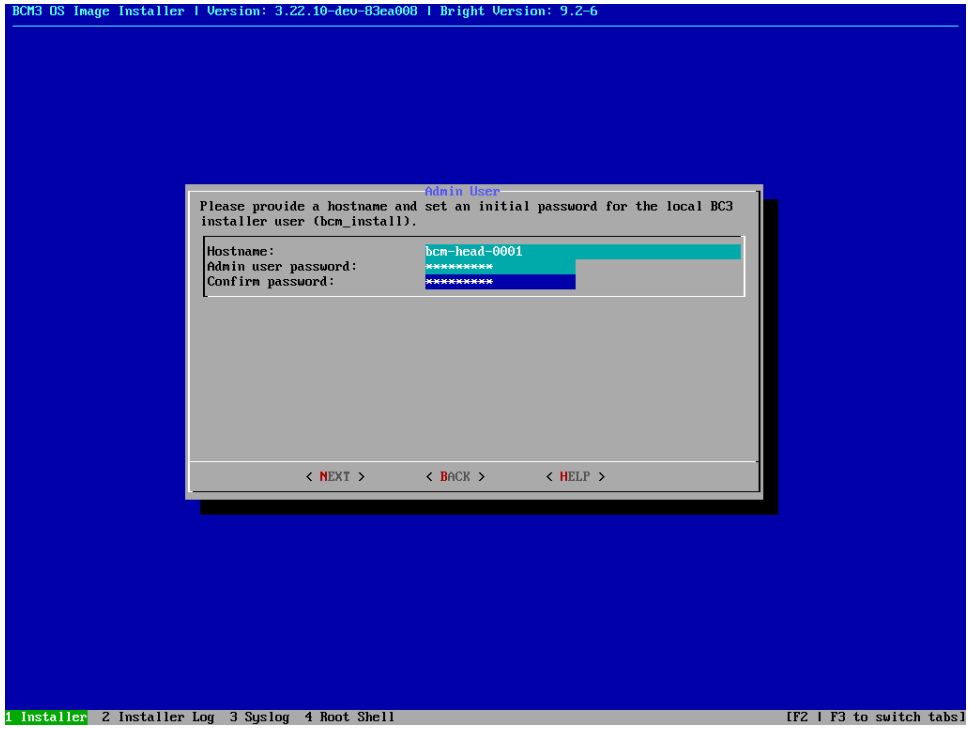
4. Download the Base Command Manager installer ISO from Cloud Storage.
5. Burn the ISO to a DVD or to a bootable USB device.

   It can also be mounted as virtual media and installed using the BMC. The specific mechanism for the latter will vary by vendor.
6. Ensure that the BIOS of the target head node is configured in UEFI mode and that its boot order is configured to boot the media containing the Bright installer image.
7. Boot the installation media.
8. At the grub menu, choose `Head Node Base OS Installer`.

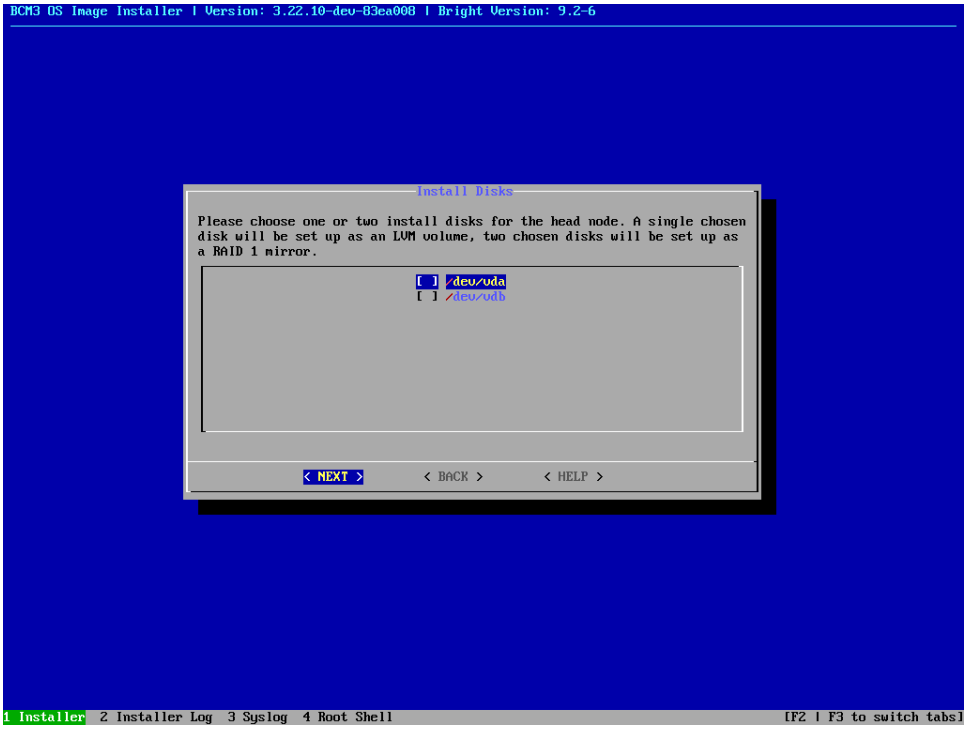
9. After booting and at the Welcome screen, press Enter to select the `Start` option and begin installation.



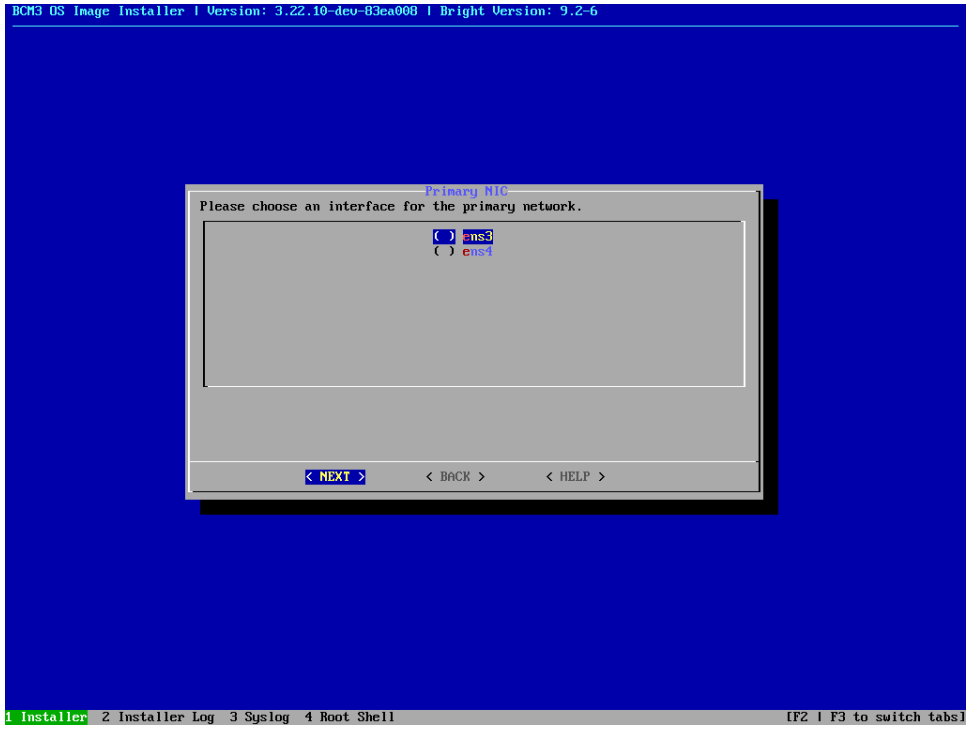
10. Confirm the hostname of the primary head node, or update it as necessary, and enter a password for the `bcm_install` user. This will be used to login to the head node after the OS is installed and complete Base Command Manager.



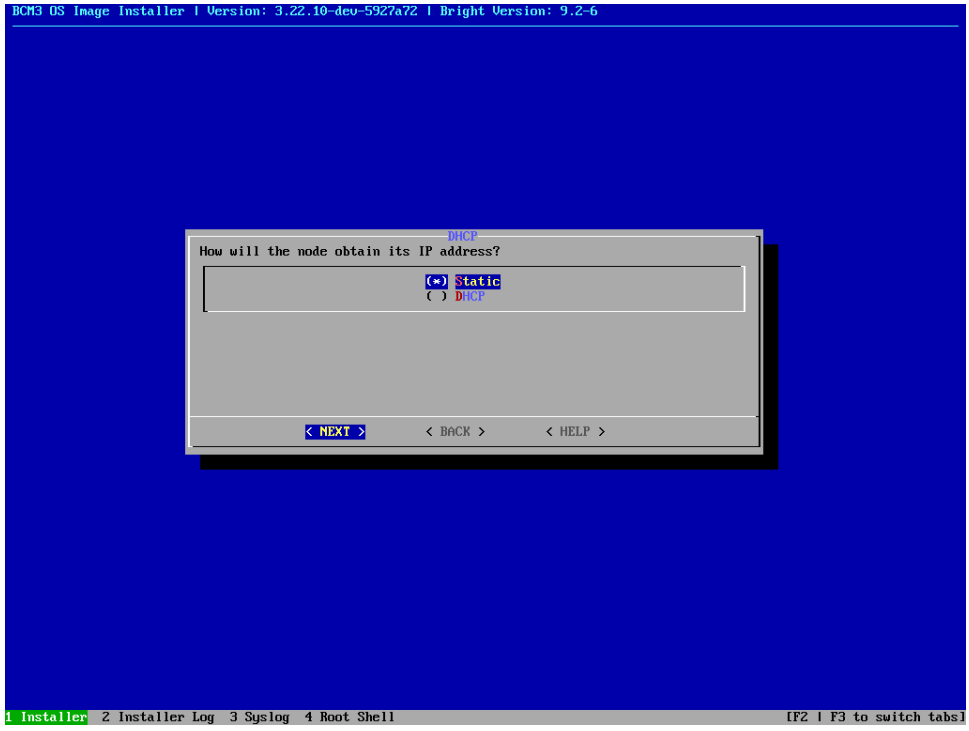
11. Select one or more disks to be used for OS installation.



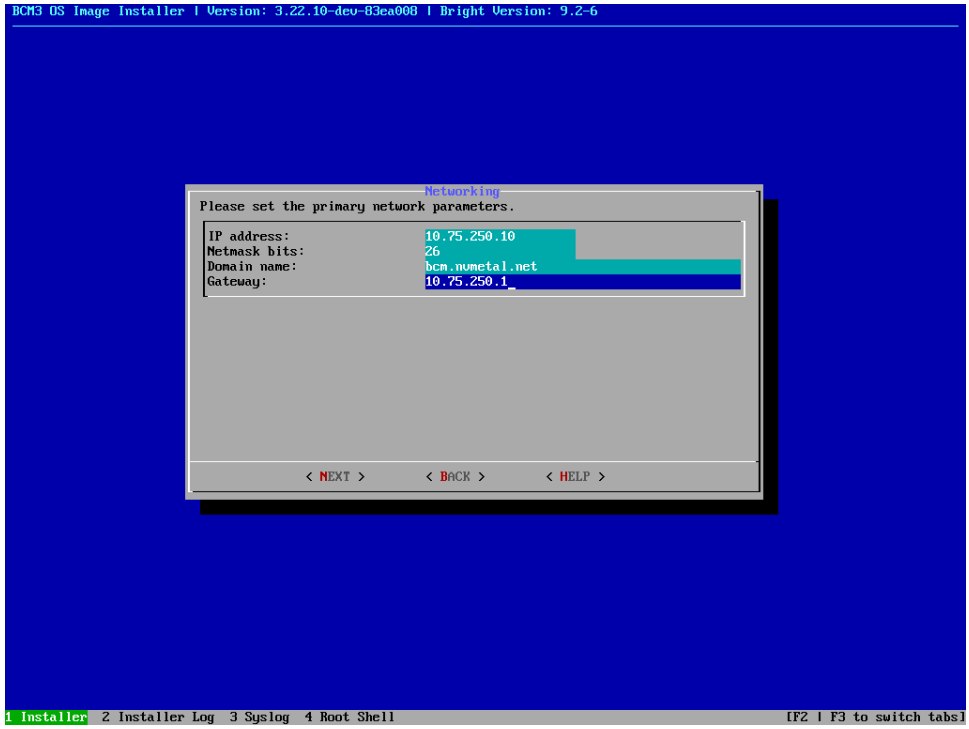
12. Choose the primary network interface for the head node. This is the `internalnet` interface and should have Internet access.



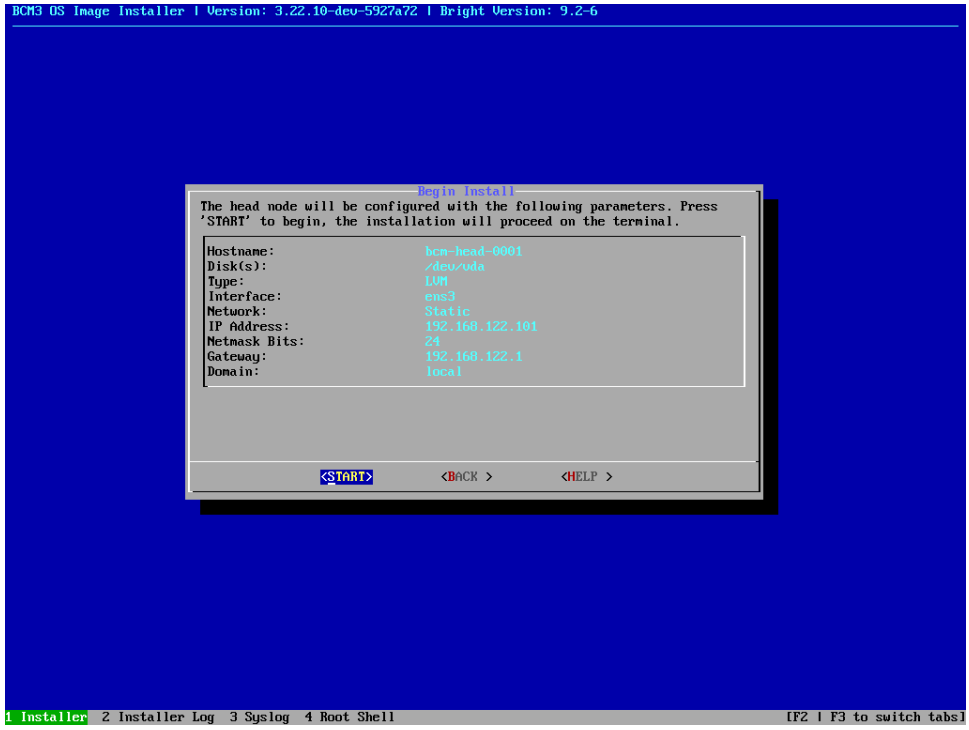
13. Specify whether the primary interface is statically configured or uses DHCP.



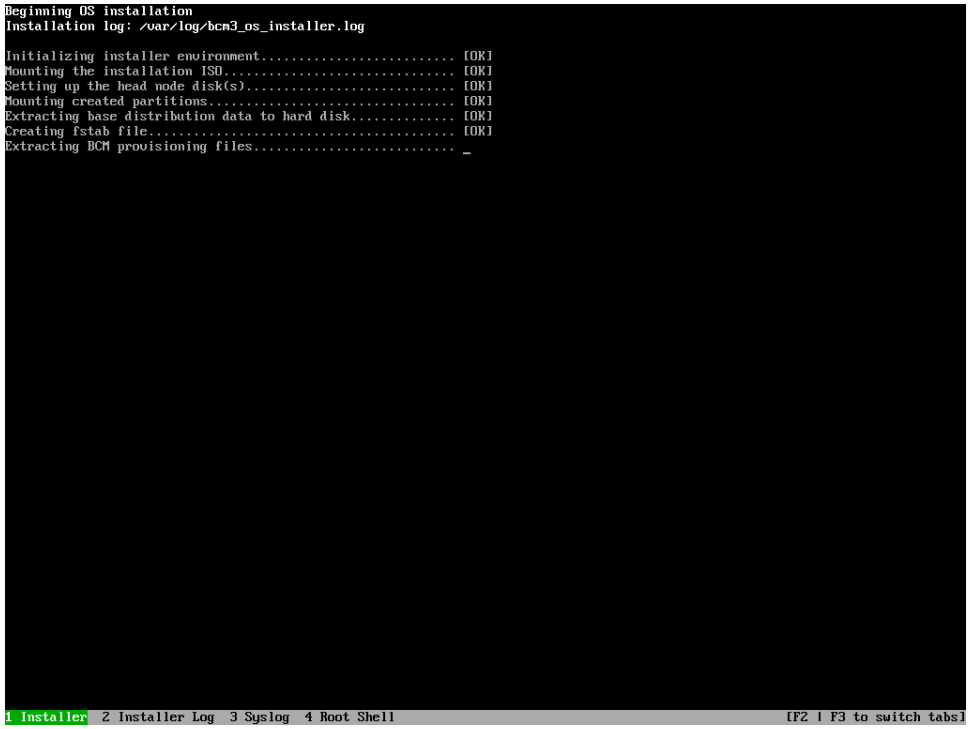
14. .If statically configured, enter the interface configuration parameters.



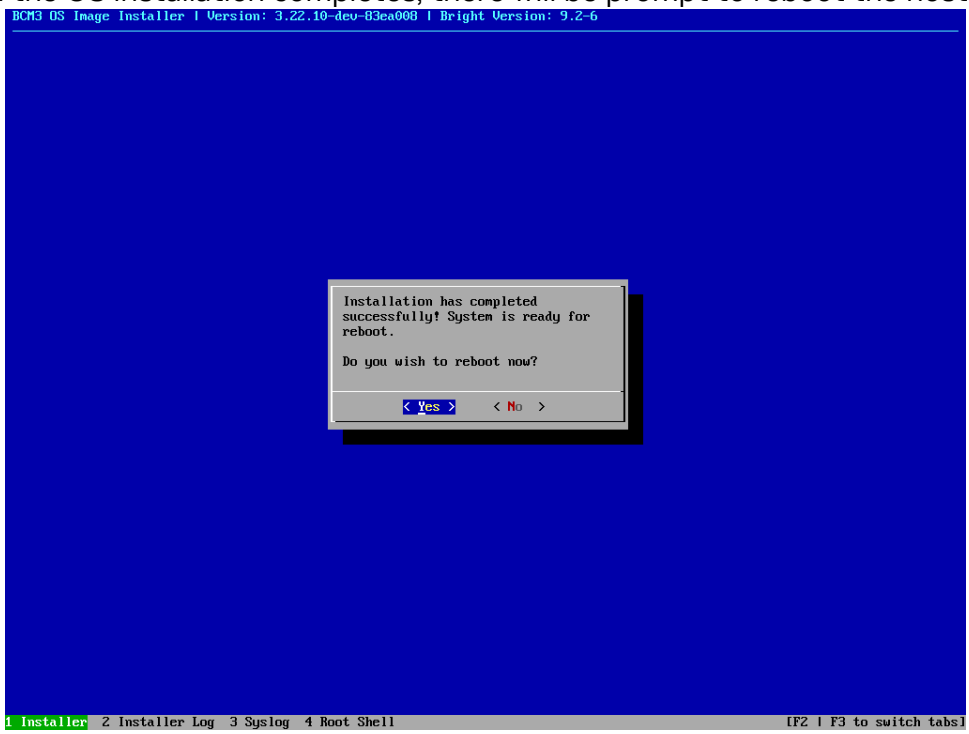
15. Confirm the settings at the summary screen and Select Start to install the OS.



16. Track the installation on the resulting screen.



17. When the OS installation completes, there will be prompt to reboot the host.



18. After the host reboots, login as the `bcm_install` user using the password provided to the OS installer. ssh can be used instead of the out-of-band console at this point.

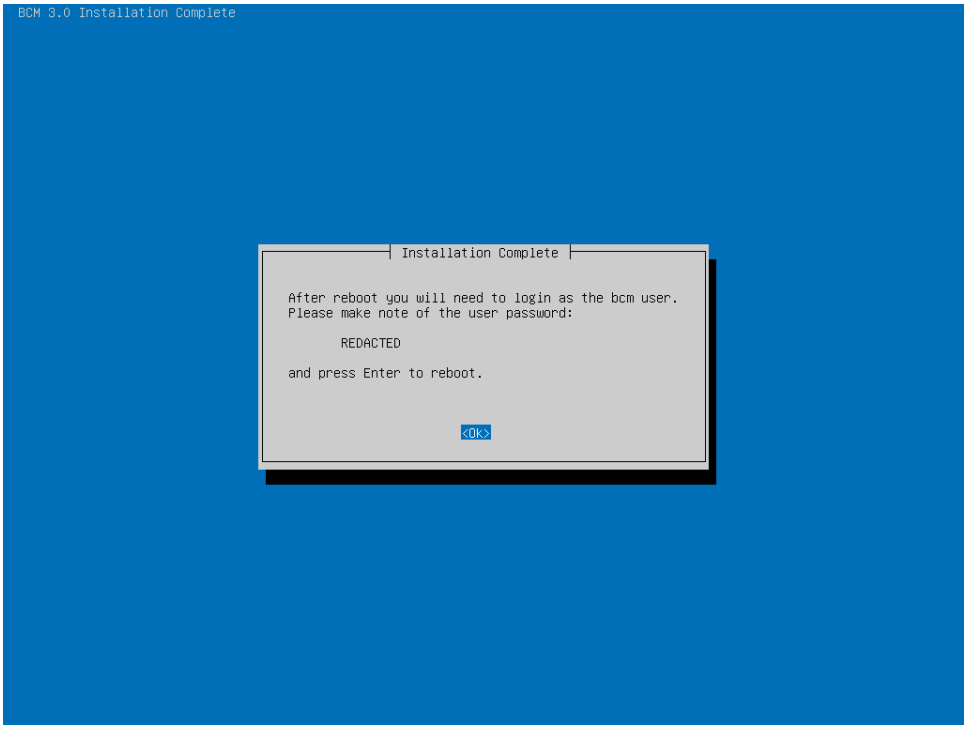
19. Run the `configure_install` command.

```
sudo /opt/bcm/configure_install
```

20. After the configuration completes, run the `install` command.

```
sudo /opt/bcm/installer/install
```

21. When installation completes, make note of the randomly generated password for the bcm admin user, and select Enter to reboot.



22. At this step there will be one DGX node and one CPU node in the device list. These hosts will not have MAC and IP assignments.

    Before proceeding, configure interfaces and IP addresses in each node category.

23. Clone the DGX nodes.

    `dgx01` was created during head node installation. Clone it to create the DGX nodes.

```
% device
% foreach --clone dgx01 -n dgx02..[dgxXX] ()
% commit
```

24. Check the nodes and their categories.

    Extra options are used for `device list` to make the format more readable.

```
% device list -f hostname:20,category:10,ip:20,status:15
hostname (key)       category       ip                   status
-------------------- ----------     -------------------- ---------------
bcm-head-01                        10.130.122.254       [   UP   ]
dgx01                dgx            10.130.122.5         [  DOWN  ]
dgx02                dgx            10.130.122.6         [  DOWN  ]
dgx03                dgx            10.130.122.7         [  DOWN  ]
dgx04                dgx            10.130.122.8         [  DOWN  ]
```

25. License cluster by running the request-license and providing product key.

```
request-license
Product Key (XXXXXX-XXXXXX-XXXXXX-XXXXXX-XXXXXX):
```

2 Head Node ConfigurationConfigure Bright to Allow MAC Addresses to PXE Boot

1. Use the root (not `cmsh`) shell.
2. In `/cm/local/apps/cmd/etc/cmd.conf`, uncomment the `AdvancedConfig` parameter.
   ```
   AdvancedConfig = { "DeviceResolveAnyMAC=1" } # modified value
   ```
3. Restart the `CMDaemon` to enable reliable PXE booting from bonded interfaces.
   ```
   # systemctl restart cmd
   ```

The `cmsh` session will be disconnected because of restarting the `CMDaemon`. Type `connect` to reconnect after the `CMDaemon` has restarted. Or enter `exit` and then restart `cmsh`.

2.2 Configure Network Interfaces on the DGX Nodes

The steps that follow are performed on the head node and should be run on all DGX systems.

Note: Double check the MAC address for each interface and the IP number for the `bond0` interface. Mistakes here will be difficult to diagnose.

1. Set the MAC addresses on the physical interfaces.

```
# cmsh
% device
% use dgx01
% interfaces
% use ipmi0
% set ip 10.130.111.68
% set gateway 10.130.111.65
% use enp225s0f1np1
% set mac B8:CE:F6:2F:08:69
% use enp97s0f1np1
% set mac B8:CE:F6:2D:0E:A7..
% % commit

% list
Type         Network device name  IP               Network          Start if
------------ -------------------- ---------------- ---------------- --------
bmc          ipmi0                10.130.111.68    ipminet          always
bond         bond0 [prov]         10.130.122.5     internalnet      always
physical     enp225s0f1 (bond0)   0.0.0.0                           always
physical     enp97s0f1 (bond0)    0.0.0.0                           always
```

2. Verify the configuration.

```
% get provisioninginterface
bond0
% interfaces
% list
Type         Network device name  IP               Network          Start if
------------ -------------------- ---------------- ---------------- --------
bmc          ipmi0                10.130.111.68    ipminet          always
bond         bond0 [prov]         10.130.122.5     internalnet      always
physical     enp225s0f1np1 (bond0)   0.0.0.0                       always
physical     enp97s0f1np1 (bond0)    0.0.0.0                       always
```

3. Configure InfiniBand interfaces on DGX Nodes

   The following procedure adds four physical InfiniBand interfaces for a single DGX system (`dgx01`).

```
% /      # go to top level of CMSH
% device
% use dgx01
% interfaces
% add physical ibp12s0
% set ip 10.149.0.5
% set network ibnet
% add physical ibp75s0
% set ip 10.149.1.5
% set network ibnet
% add physical ibp141s0
% set ip 10.149.2.5
% set network ibnet
% add physical ibp186s0
% set network ibnet
% set ip 10.149.3.5

% list
Type         Network device name  IP               Network          Start if
```

```
------------ -------------------- ---------------- ----------------- --------
bmc          ipmi0                10.130.111.69    ipminet           always
bond         bond0 [prov]         10.130.122.5     internalnet       always
physical     enp225s0f1np1 (bond0)   0.0.0.0                        always
physical     enp97s0f1np1 (bond0)    0.0.0.0                        always
physical     ibp12s0              10.149.0.5       ibnet             always
physical     ibp141s0             10.149.2.5       ibnet             always
physical     ibp186s0             10.149.3.5       ibnet             always
physical     ibp75s0              10.149.1.5       ibnet             always

% device commit
```

2.3 Identify the DGX Cluster Nodes

1. Identify the nodes by setting the MAC address for the provisioning interface for each node to the MAC address listed in the site survey.

```
% device
% use dgx01
% set mac b8:ce:f6:2f:08:69
% use dgx02
% set mac 0c:42:a1:54:32:a7
% use dgx03
% set mac 0c:42:a1:0a:7a:51
% use dgx04
% set mac 1c:34:da:29:17:6e
% foreach -c dgx (get mac)
B8:CE:F6:2F:08:69
0C:42:A1:54:32:A7
0C:42:A1:0A:7A:51
1C:34:DA:29:17:6E
```

2. If all the MAC addresses are set properly, commit the changes.

```
% device commit
% quit
```

2.4 Identify the First CPU Node

1. Set the IP address for the IPMI interface.

```
% device
% use bcm-cpu-01
% interfaces
% use ipmi0
% set ip 10.127.1.15
% set gateway 10.127.1.1
% commit
```

2. Set the MAC addresses for the Ethernet interfaces.

```
% device
% use bcm-cpu-01
% interfaces
% use ens2f0np0
% set mac 88:e9:a4:92:26:ba
% use ens2f1np1
% set mac 88:e9:a4:92:26:bb
% commit
```

3. Set the IP address for the `bond0` interface.

```
% device
% use bcm-cpu-01
% interfaces
% use bond0
% set ip 10.127.3.15
% commit
```

2.5 Power On and Provision the Cluster Nodes

Now that the required post-installation configuration has been completed, it is time to power on and provision the cluster nodes. After the initial provisioning, power control will be available from within Bright using the `cmsh` or Bright View. But for this initial provisioning it is necessary to power them on outside of Bright (that is, using the power button or a KVM).

It will take several minutes for the nodes to go through their BIOS. After that, you should see the node status progress as the nodes are being provisioned. Watching the `/var/log/messages` and `/var/log/node-installer` log files to verify that everything is proceeding smoothly

3 High Availability

This section covers how to configure high availability (HA) using `cmha-setup` CLI wizard.

1. Ensure that both head nodes are licensed.

   The MAC address for the secondary head was provided when the cluster license was installed.

```
% main licenseinfo  | grep ^MAC
MAC address / Cloud ID               04:3F:72:E7:67:07|14:02:EC:DA:AF:18
```

2. Configure the shared storage (NFS).

   Mounts configured in `fsmounts` will be automatically mounted by the `CMDaemon`.

```
% device
% use master
% fsmounts
% add /nfs/general
% set device 10.130.122.252:/var/nfs/general
% set filesystem nfs
% commit
% show
Parameter                      Value
------------------------------ ------------------------------------------------
Device                         10.130.122.252:/var/nfs/general
Revision
Filesystem                     nfs
Mountpoint                     /nfs/general
Dump                           no
RDMA                           no
Filesystem Check               NONE
Mount options                  defaults
```

3. Verify that the shared storage is mounted.

```
# mount | grep '/nfs/general'
10.130.122.252:/var/nfs/general on /nfs/general type nfs4
(rw,relatime,vers=4.2,rsize=1048576,wsize=1048576,namlen=255,hard,proto=tcp,timeo
=600,retrans=2,sec=sys,clientaddr=10.130.122.254,local_lock=none,addr=10.130.122.
252)
```

4. Verify that head node has power control over the cluster nodes.

```
% device
% power -c dgx status
[-head1->device]% power -c dgx status
ipmi0 ..................... [    ON     ] dgx01
ipmi0 ..................... [    ON     ] dgx02
ipmi0 ..................... [    ON     ] dgx03
ipmi0 ..................... [    ON     ] dgx04
[bcm-head-01->device]%
```

5. Power off the cluster nodes.

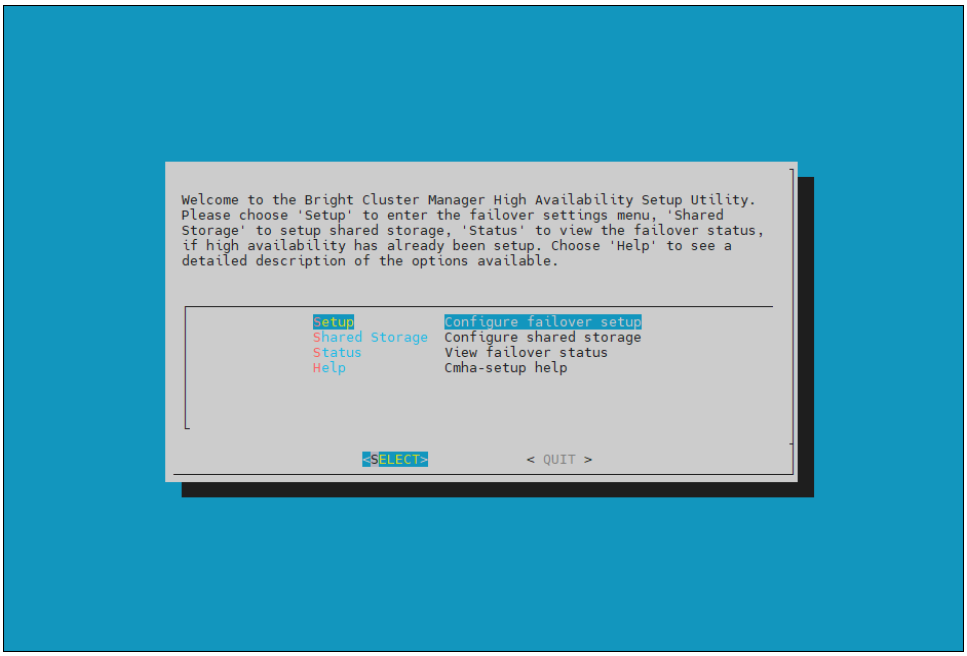
   The cluster nodes must be powered off before configuring HA.

```
% power -c dgx off
ipmi0 ..................... [    OFF    ] dgx01
ipmi0 ..................... [    OFF    ] dgx02
ipmi0 ..................... [    OFF    ] dgx03
ipmi0 ..................... [    OFF    ] dgx04
```

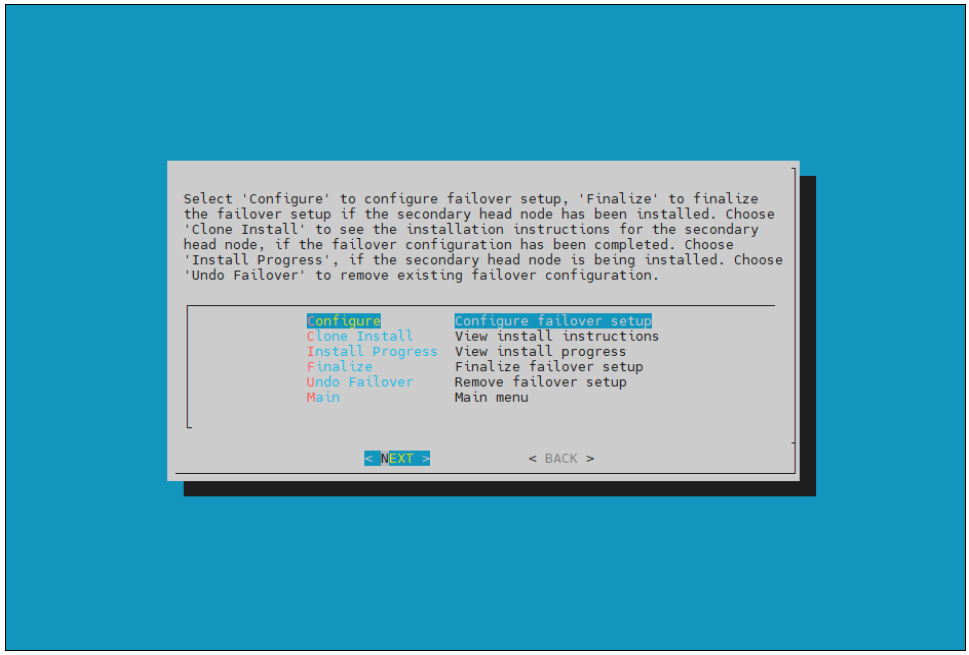
6. Start the `cmha-setup` CLI wizard as the root user on the primary head node.

```
# cmha-setup
```

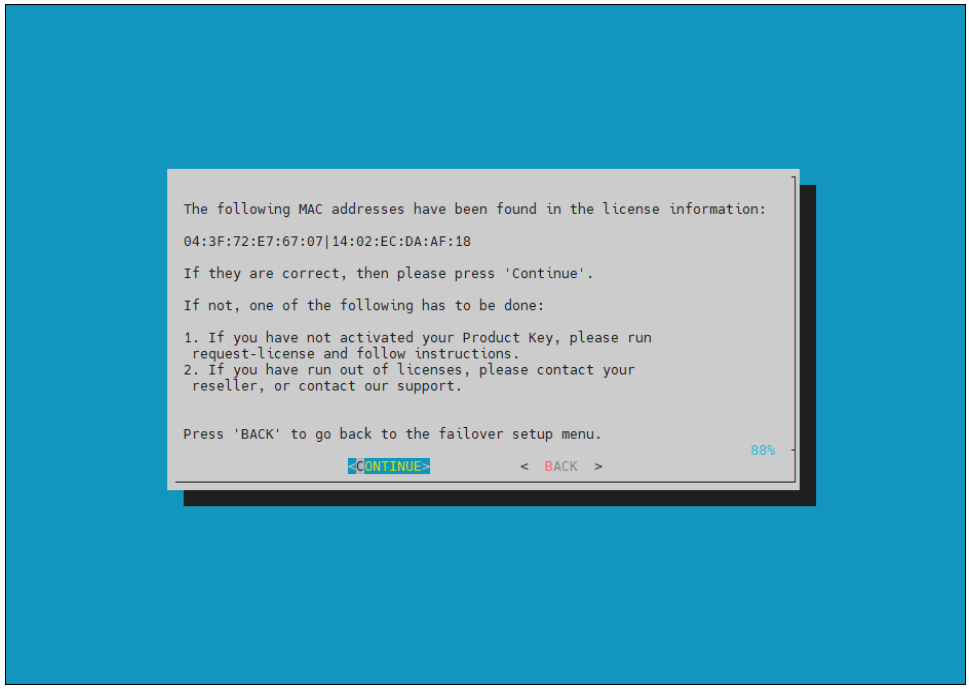
7. Select `Setup`.



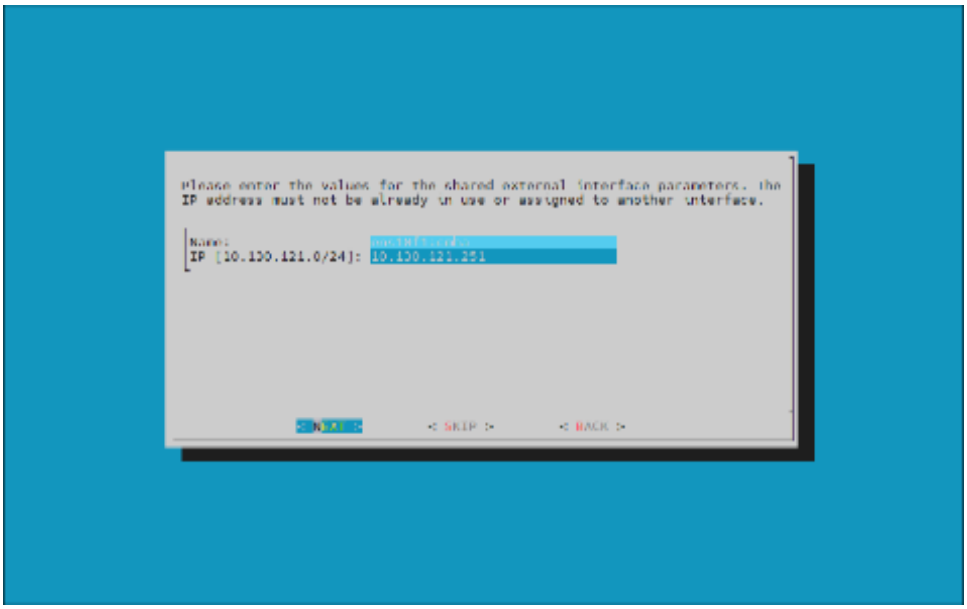
8. Select `Configure`.



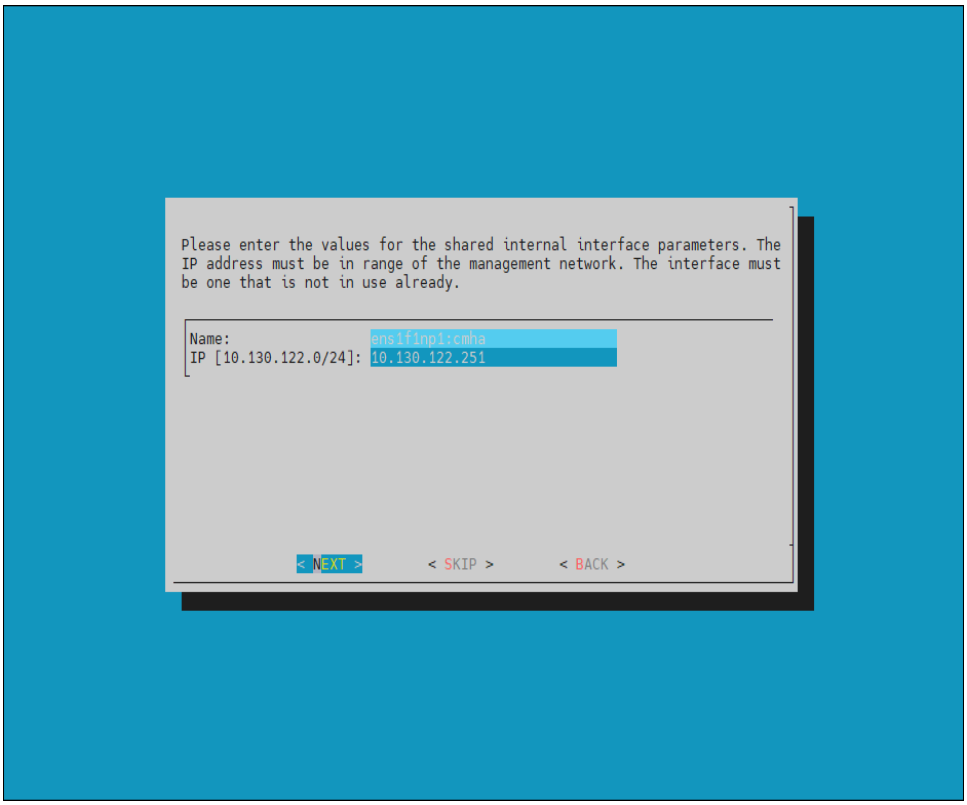
9. Verify that the cluster license information found `cmha-setup` is correct.



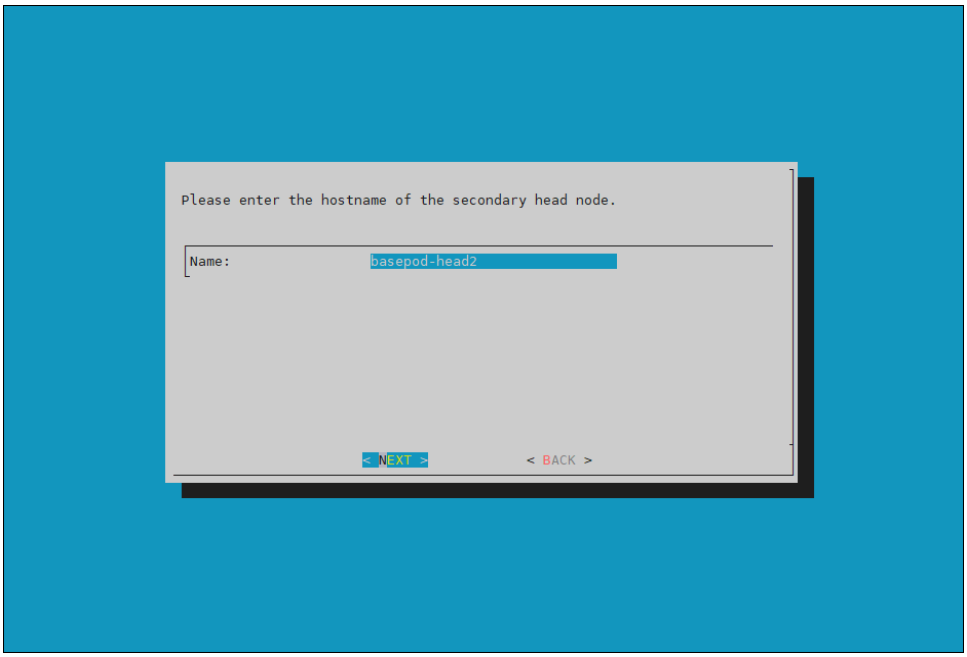
10. Configure an external Virtual IP address that will be used by the active head node in the HA configuration. (This will be the IP that should always be used for accessing the active head nodes.)



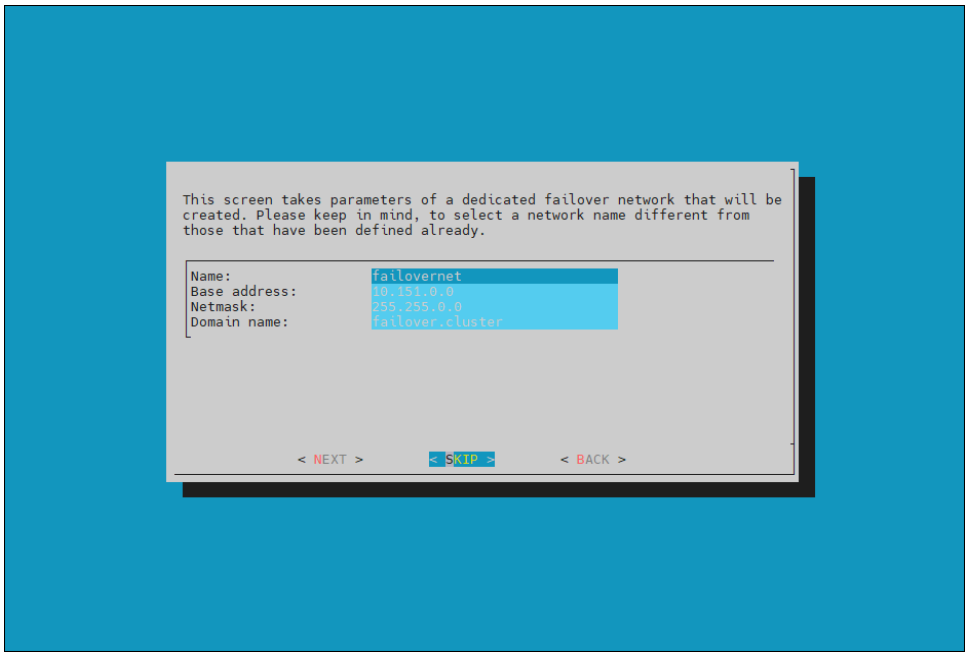
11. Provide an internal Virtual IP address that will be used by the active head node in the HA configuration.



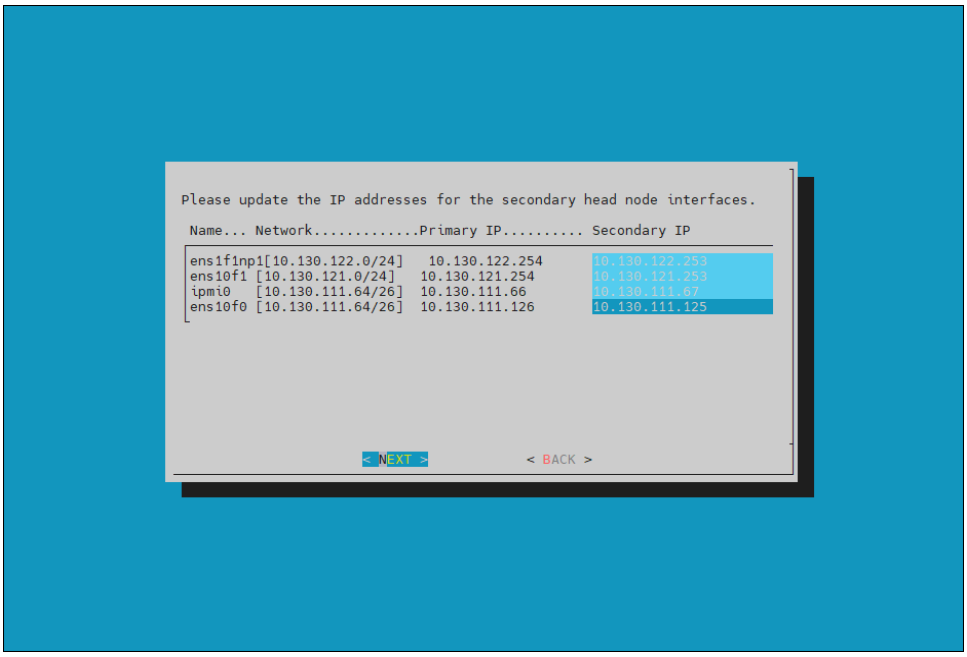
12. Provide the name of the secondary head node.



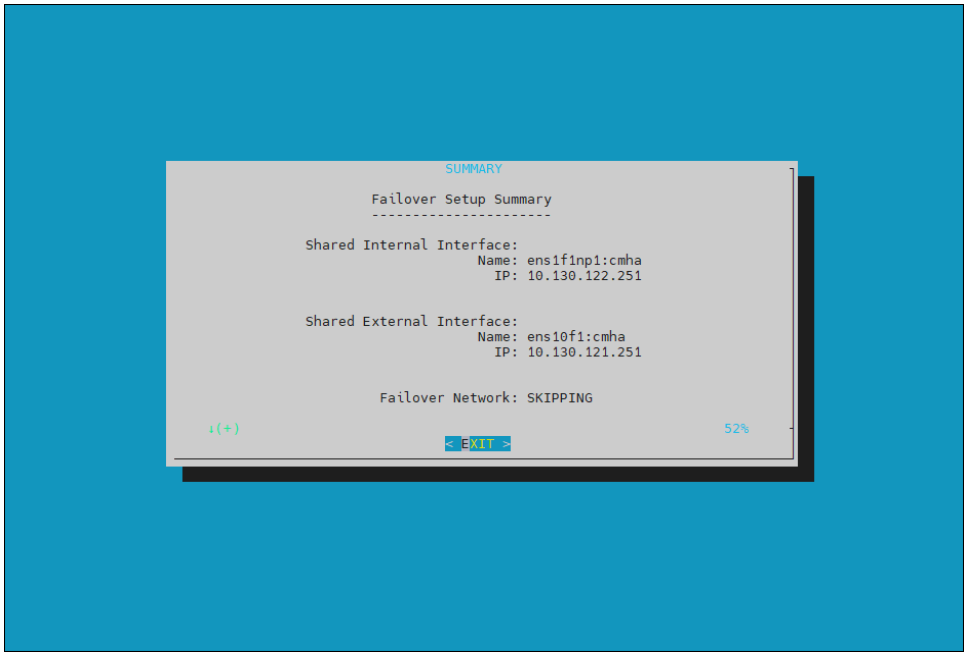
13. DGX SuperPOD uses the internal network as the failover network, so select `SKIP` to continue.



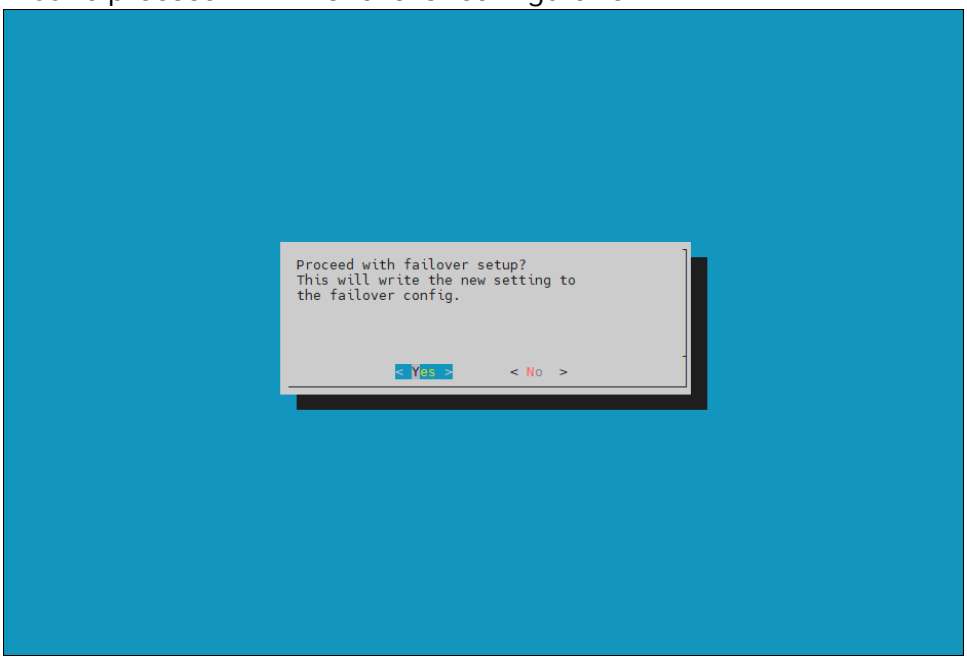
14. Configure the IP addresses for the secondary head node that the wizard is about to create.



15. The wizard shows a summary of the information that it has collected. The VIP that will be assigned to the internal and external interfaces, respectively.

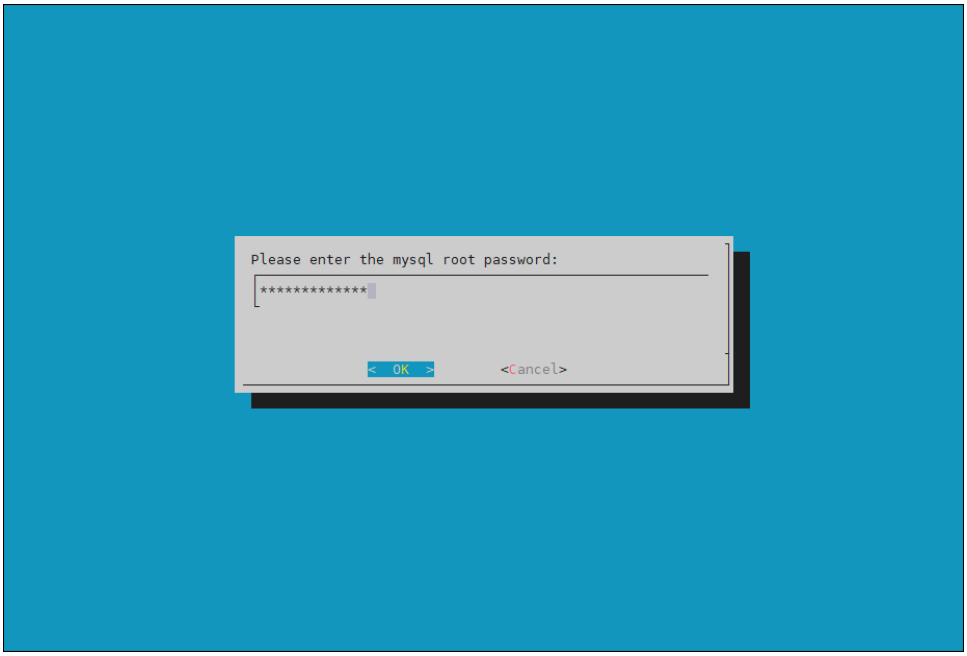


16. Select `Yes` to proceed with the failover configuration.



17. Enter the MySQL root password.

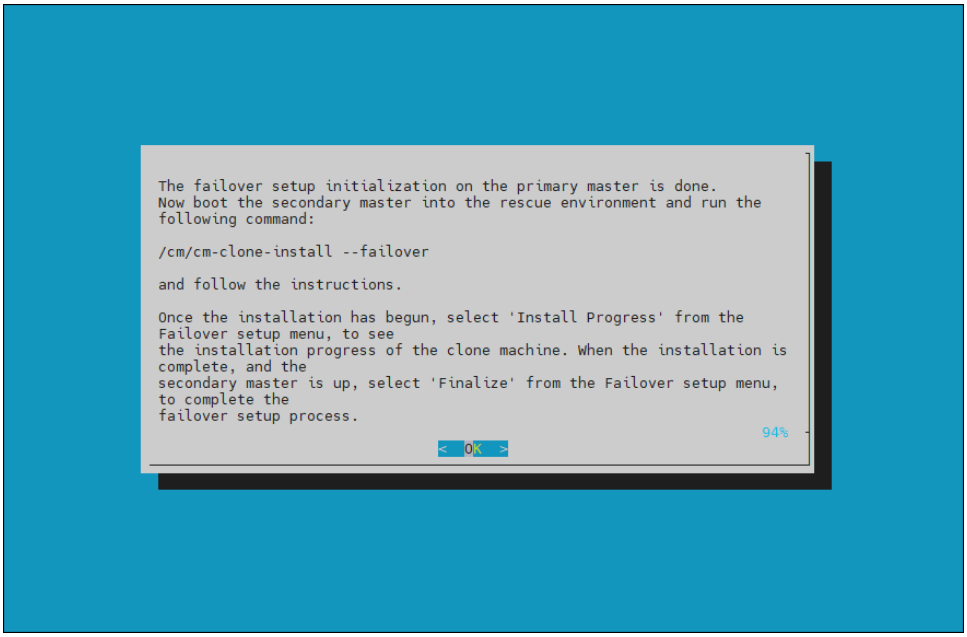
    The auto-generated password is in `/root/.mysql`.



18. The wizard implements the first steps in the HA configuration. If all the steps show `OK`, press `ENTER` to continue. The progress is shown below.

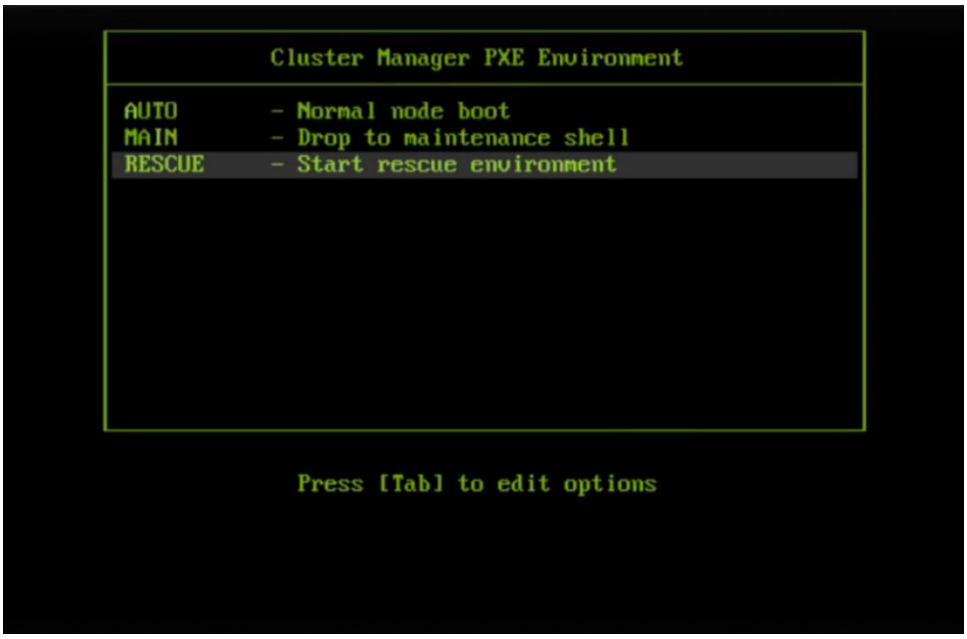
```
Initializing failover setup on master..... [  OK  ]
Updating shared internal interface........ [  OK  ]
Updating shared external interface........ [  OK  ]
Updating extra shared internal interfaces. [  OK  ]
Cloning head node......................... [  OK  ]
Updating secondary master interfaces...... [  OK  ]
Updating Failover Object.................. [  OK  ]
Restarting cmdaemon....................... [  OK  ]
Press any key to continue
```

19. Run the `/cm/cm-clone-install -failover` command on the secondary head node. This should be a one-time network boot.



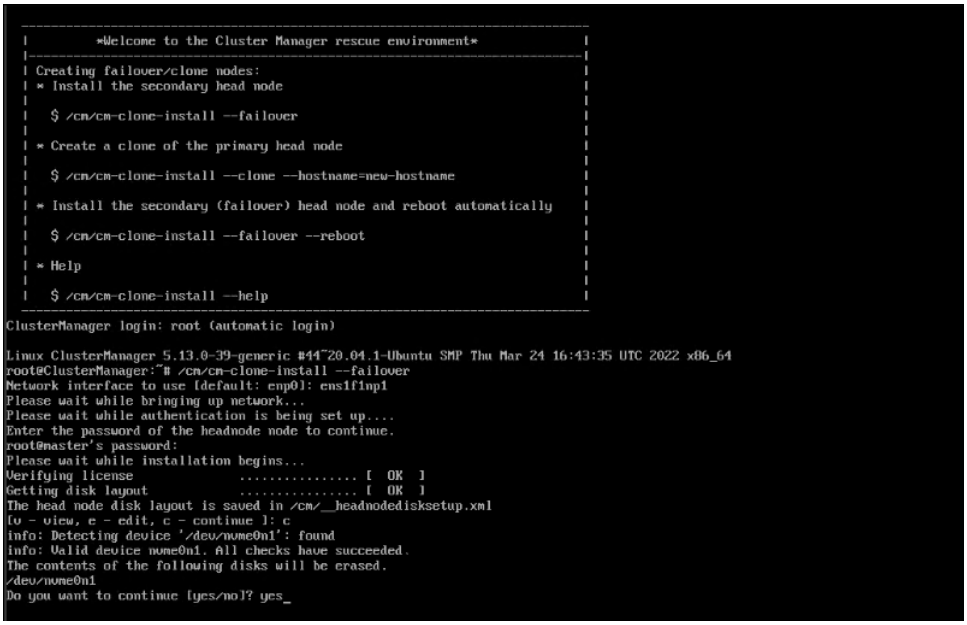
20. PXE boot the secondary head node, then select `RESCUE` from the grub menu.

    Because this is the initial boot of this node, it must be done outside of Base Command Manager (BMC or physical power button).



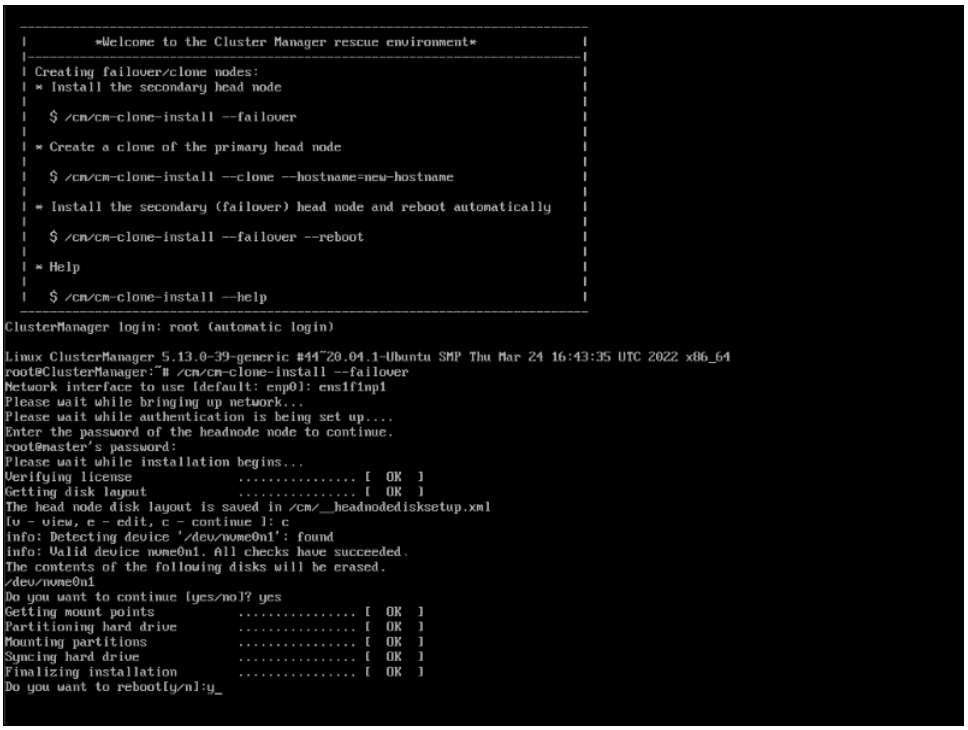
21. After the secondary head node has booted into the rescue environment, run the `/cm/cm-clone-install -failover` command, then enter yes when prompted.

    The secondary head node will be cloned from the primary.



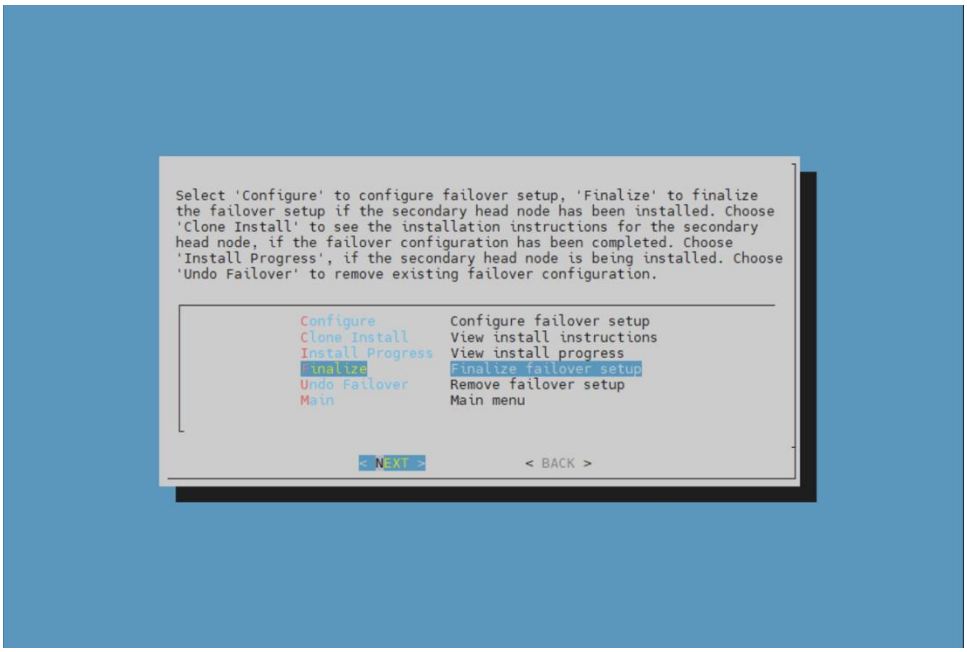
22. When cloning is completed, enter `y` to reboot the secondary head node.

    The secondary must be set to boot from its hard drive. PXE boot should not be enabled.

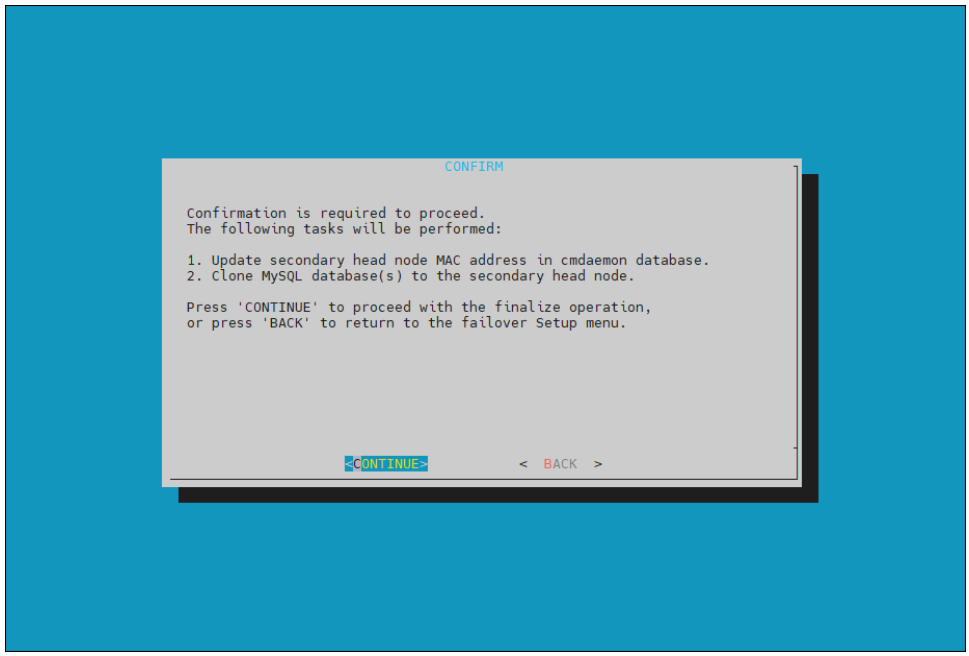


23. Wait for the secondary head node to reboot and then continue the HA setup procedure on the primary head node.
24. Select `finalize` from the `cmha-setup` menu.

    This will clone the MySQL database from the primary to the secondary head node.

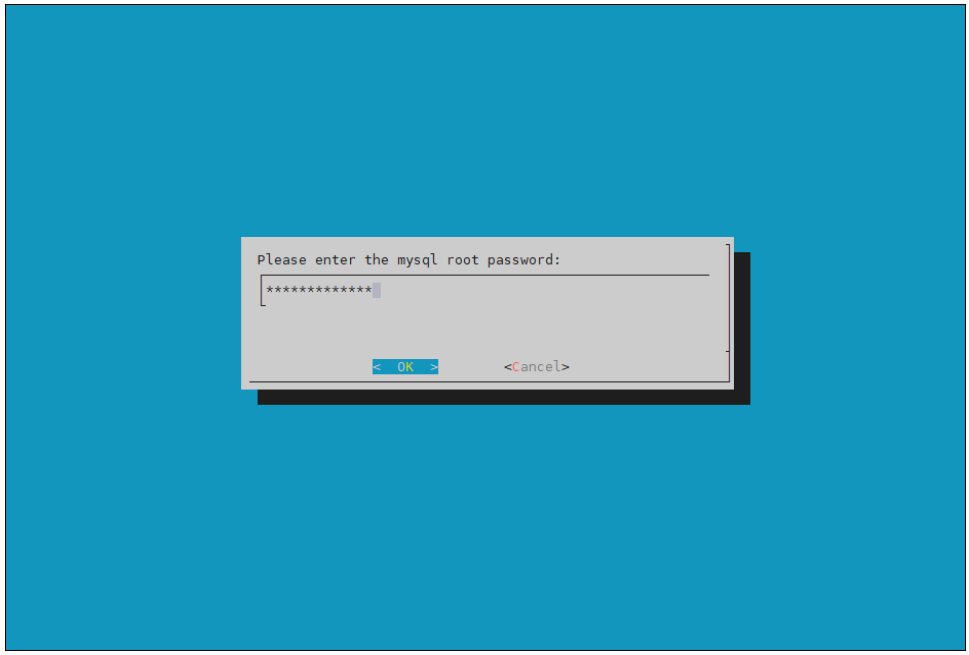


25. Select <CONTINUE> on the confirmation screen.

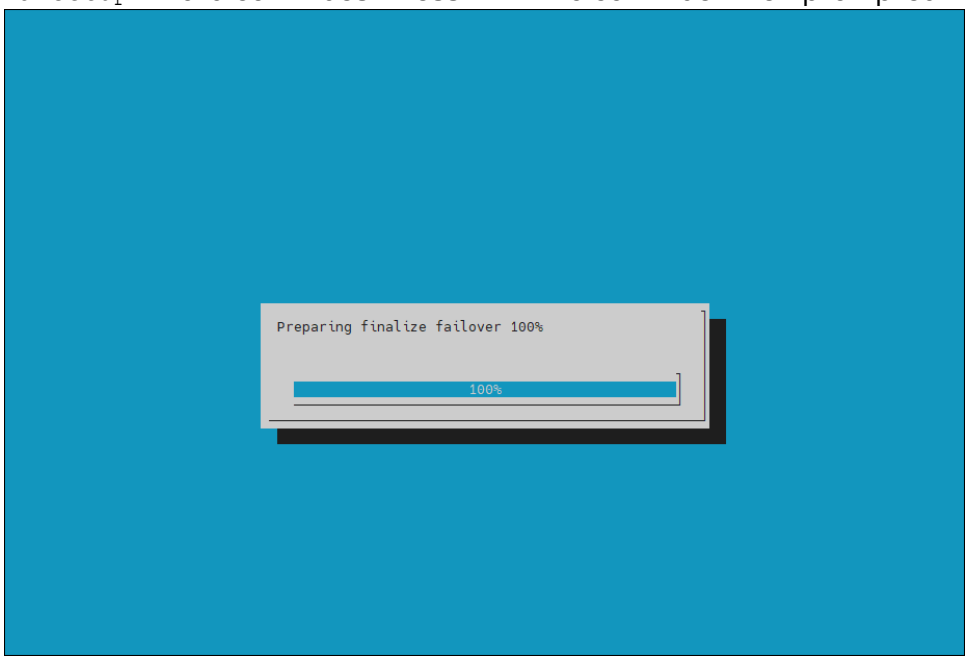


26. Enter the MySQL root password.

    The auto-generated password is in `/root/.mysql`.



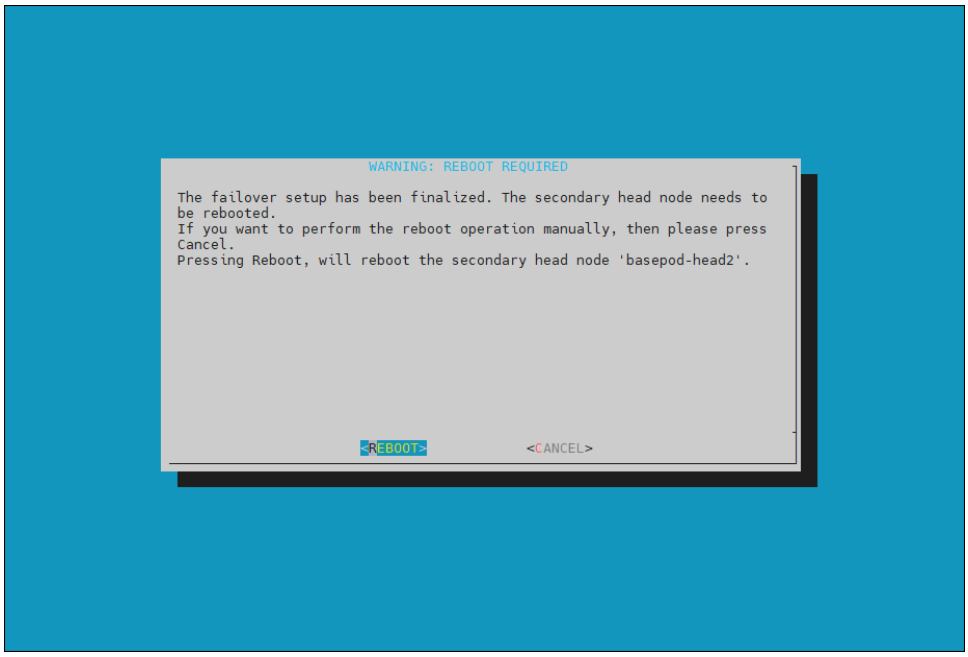
27. The `cmha-setup` wizard continues. Press `ENTER` to continue when prompted.



The progress is shown below:

```
Updating secondary master mac address.............. [  OK  ]
Initializing failover setup on bcm-head-02....... [  OK  ]
Stopping cmdaemon.................................. [  OK  ]
Cloning cmdaemon database.......................... [  OK  ]
Checking database consistency...................... [  OK  ]
Starting cmdaemon, chkconfig services.............. [  OK  ]
Cloning workload manager databases................. [  OK  ]
Cloning additional databases....................... [  OK  ]
Update DB permissions.............................. [  OK  ]
Checking for dedicated failover network............ [  OK  ]
Press any key to continue
```

28. The `Finalize` step is now completed. Select `<REBOOT>` and wait for the secondary head node to reboot.

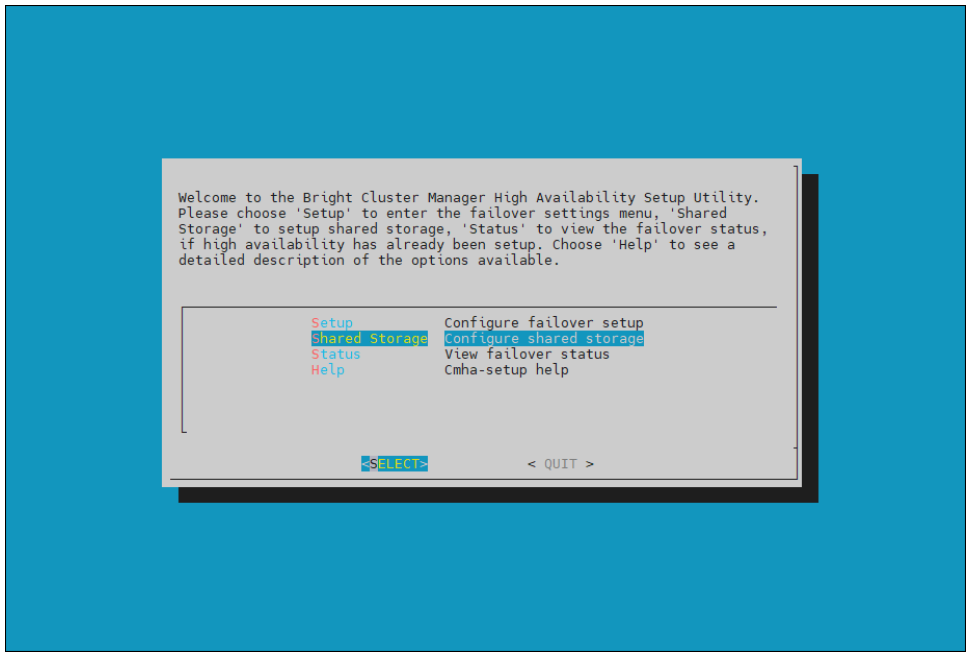


29. The secondary head node is now `UP`.

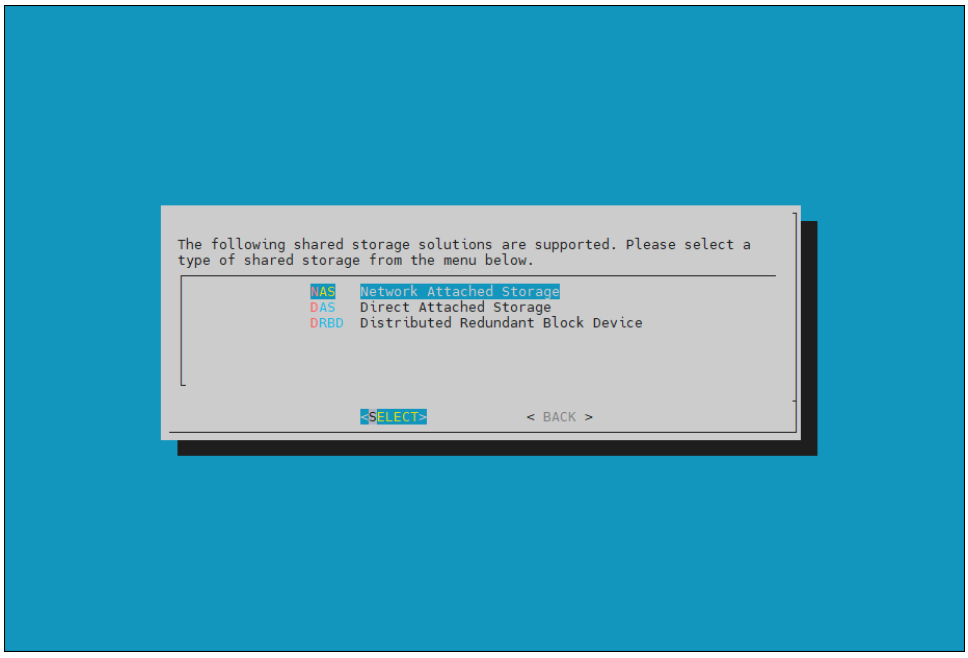
```
% device list -f hostname:20,category:12,ip:20,status:15
hostname (key)       category     ip                   status
-------------------- ---------- -------------------- ---------------
bcm-head-01                     10.130.122.254       [   UP   ]
bcm-head-02                     10.130.122.253       [   UP   ]
dgx01                dgx          10.130.122.5         [  DOWN  ]
dgx02                dgx          10.130.122.6         [  DOWN  ]
dgx03                dgx          10.130.122.7         [  DOWN  ]
dgx04                dgx          10.130.122.8         [  DOWN  ]
```

30. Select `Shared Storage` from the `cmha-setup` menu.

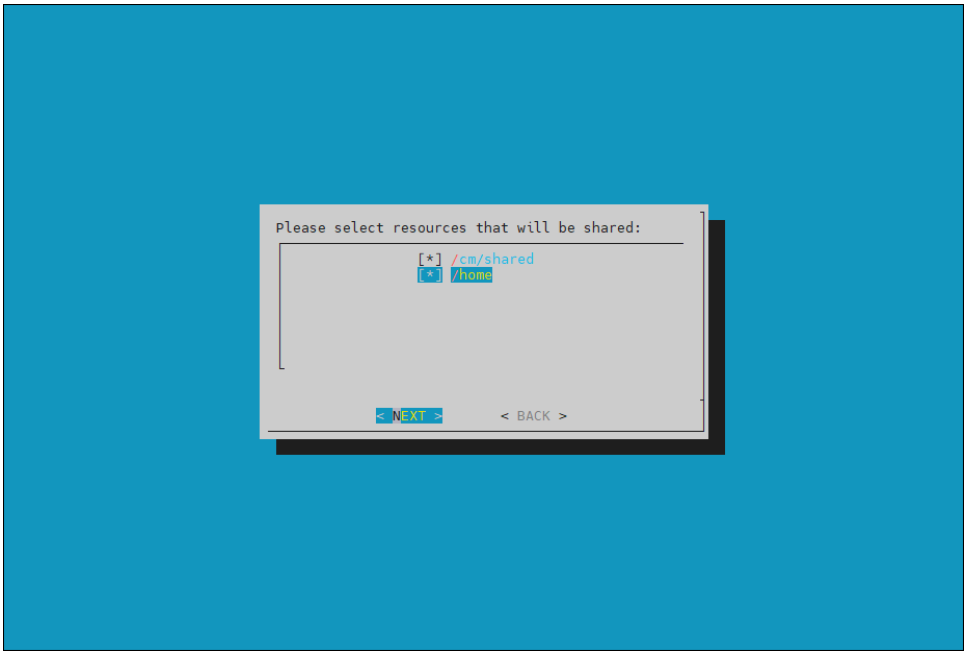
    In this final HA configuration step, `cmha-setup` will copy the `/cm/shared` and `/home` directories to the shared storage, and it configures both head nodes and all cluster nodes to mount it.



31. Select `NAS`.

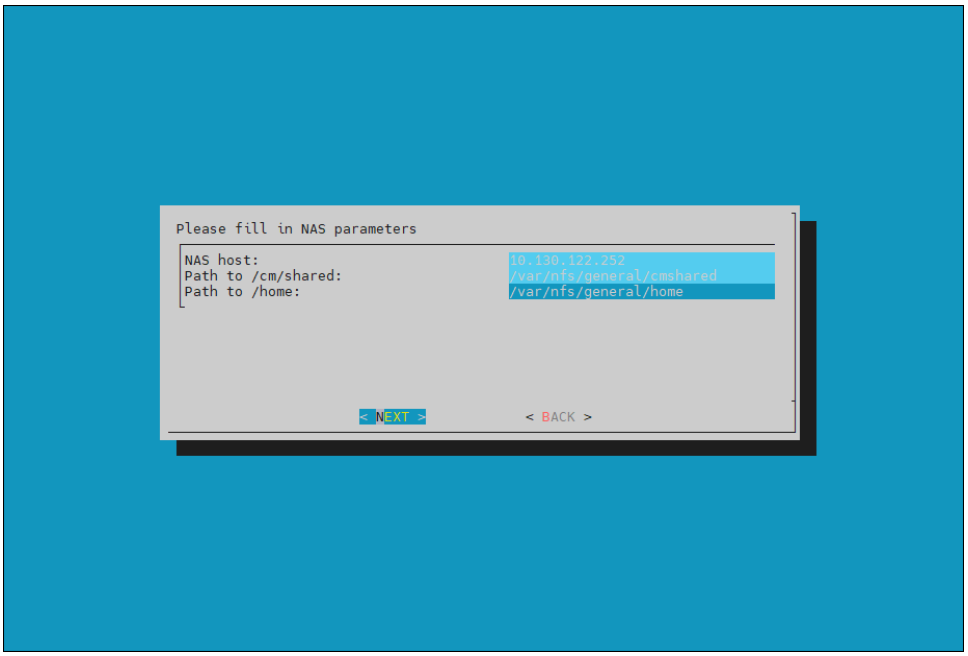


32. Select both `/cm/shared` and `/home`.



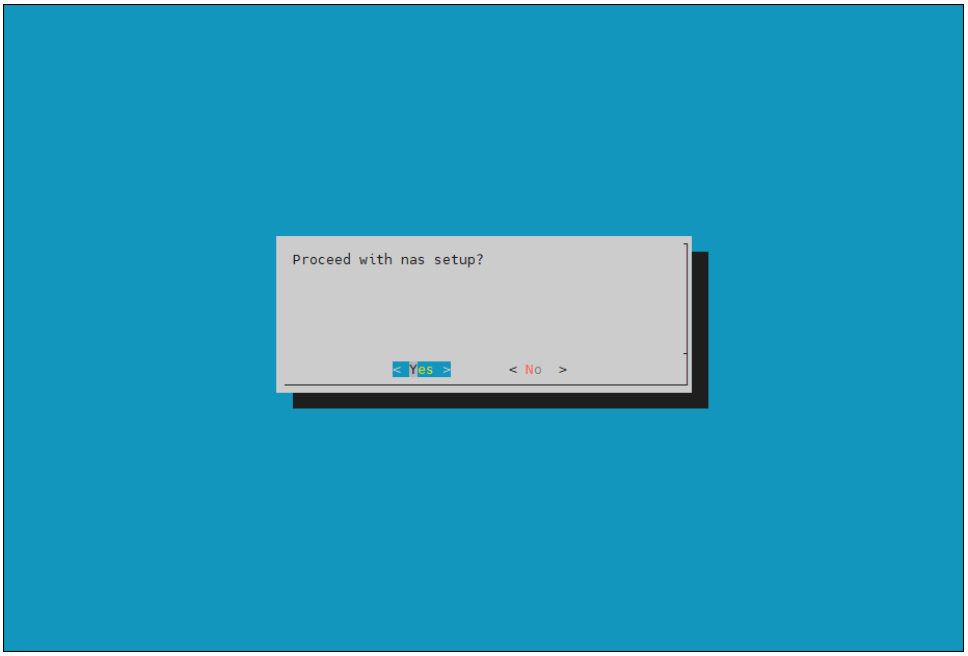
33. Provide the IP number of the NAS host, and the path that the `/cm/shared` and `/home` directories should be copied to on the shared storage.

    In this case, /`var/nfs/general` is exported, so the `/cm/shared` directory will be copied to `10.130.122.252:/var/nfs/general/cmshared`, and it will be mounted over `/cm/shared` on the cluster nodes.



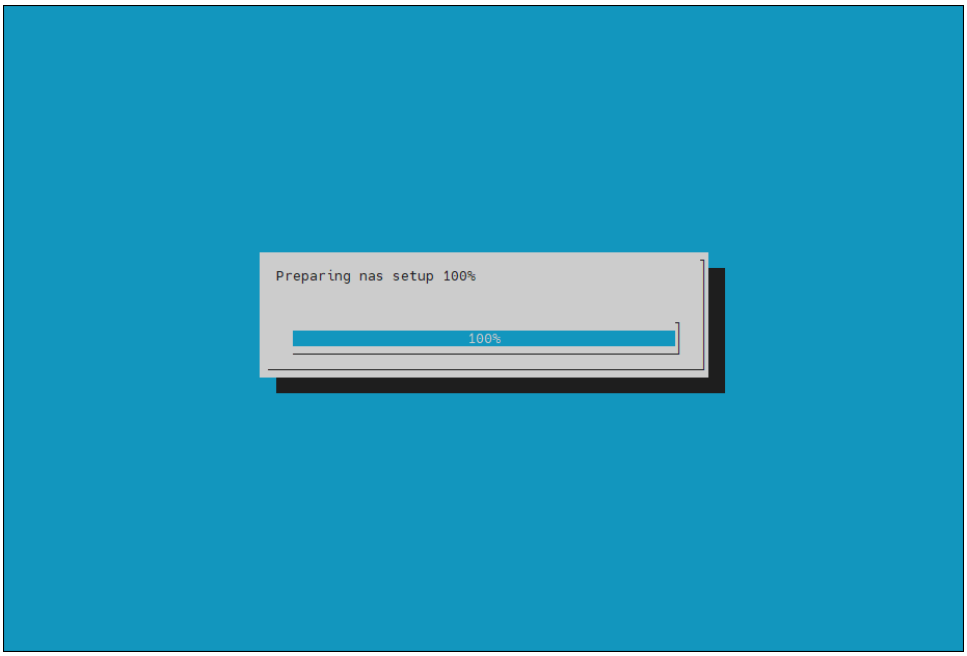
34. The wizard shows a summary of the information that it has collected. Press `ENTER` to continue.
35. Select `yes` to continue.

    This will initiate a copy and update to `fsexports`.



36. The `cmha-setup` wizard proceeds with its work.

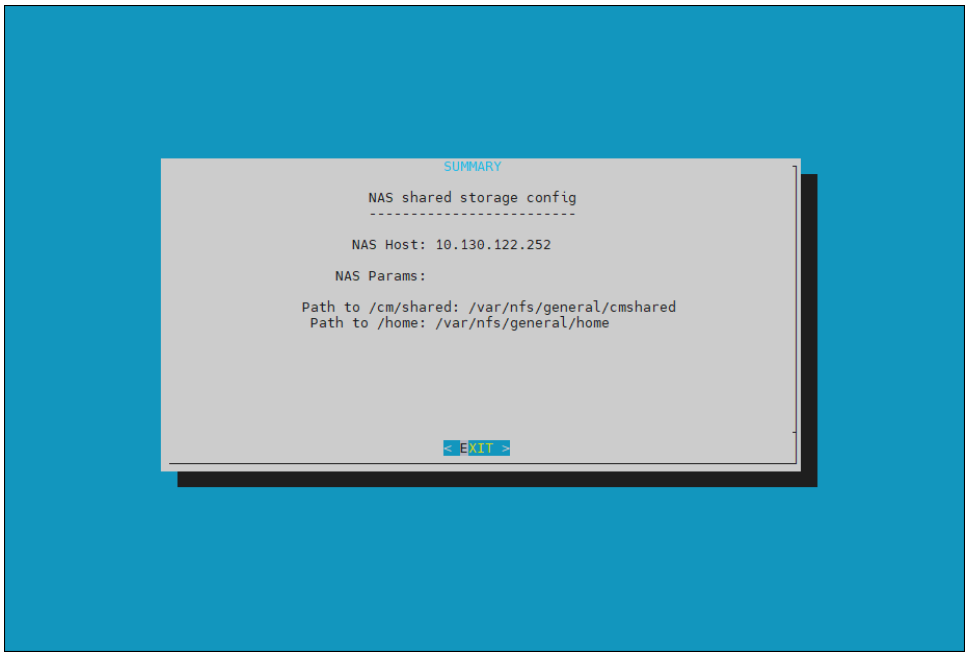
    When it completes, select `ENTER` to finish HA setup.



The progress is shown below:

```
Copying NAS data..................................... [  OK  ]
Mount NAS storage.................................... [  OK  ]
Remove old fsmounts.................................. [  OK  ]
Add new fsmounts..................................... [  OK  ]
Remove old fsexports................................. [  OK  ]
Write NAS mount/unmount scripts...................... [  OK  ]
Copy mount/unmount scripts........................... [  OK  ]
Press any key to continue
```

37. `cmha-setup` is now complete. `EXIT` the wizard to return to the shell prompt.



38. Run the `cmsh status` command to verify that the failover configuration is correct and working as expected.

    Note that the command tests the configuration from both directions: from the primary head node to the secondary, and from the secondary to the primary. The active head node is indicated by an asterisk.

```
# cmha status
Node Status: running in active mode

bcm-head-01* -> bcm-head-02
  failoverping  [  OK  ]
  mysql         [  OK  ]
  ping          [  OK  ]
  status        [  OK  ]

bcm-head-02 -> bcm-head-01*
  failoverping  [  OK  ]
  mysql         [  OK  ]
  ping          [  OK  ]
  status        [  OK  ]
```

39. Verify that the `/cm/shared` and `/home` directories are being mounted from the NAS server.

```
# mount
. . . some output omitted . . .
10.130.122.252:/var/nfs/general/cmshared on /cm/shared type nfs4
(rw,relatime,vers=4.2,rsize=32768,wsize=32768,namlen=255,hard,proto=tcp,timeo=600
,retrans=2,sec=sys,clientaddr=10.130.122.253,local_lock=none,addr=10.130.122.252)
10.130.122.252:/var/nfs/general/home on /home type nfs4
(rw,relatime,vers=4.2,rsize=32768,wsize=32768,namlen=255,hard,proto=tcp,timeo=600
,retrans=2,sec=sys,clientaddr=10.130.122.253,local_lock=none,addr=10.130.122.252)
```

40. Login to the head node to be made active and run `cmha makeactive`.

```
# ssh bcm-head-02
# cmha makeactive
================================================================================
This is the passive head node. Please confirm that this node should
become
the active head node. After this operation is complete, the HA status of
the head nodes will be as follows:

bcm-head-02 will become active head node (current state: passive)
bcm-head-01 will become passive head node (current state: active)
================================================================================

Continue(c)/Exit(e)? c

Initiating failover.............................. [  OK  ]

bcm-head-02 is now active head node, makeactive successful
```

41. Run the `cmsh status` command again to verify that the secondary head node has become the active head node.

```
# cmha status
Node Status: running in active mode

bcm-head-02* -> bcm-head-01
  failoverping  [  OK  ]
  mysql         [  OK  ]
  ping          [  OK  ]
  status        [  OK  ]

bcm-head-01 -> bcm-head-02*
  failoverping  [  OK  ]
  mysql         [  OK  ]
  ping          [  OK  ]
  status        [  OK  ]
```

42. Manually failover back to the primary head node.

```
# ssh bcm-head-01
# cmha makeactive

================================================================================
This is the passive head node. Please confirm that this node should become
the active head node. After this operation is complete, the HA status of
the head nodes will be as follows:

bcm-head-01 will become active head node (current state: passive)
bcm-head-02 will become passive head node (current state: active)
================================================================================

Continue(c)/Exit(e)? c

Initiating failover..............................  [  OK  ]

bcm-head-01 is now active head node, makeactive successful
```

43. Run the `cmsh status` command again to verify that the primary head node has become the active head node.

```
# cmha status
Node Status: running in active mode

bcm-head-01* -> bcm-head-02
  failoverping  [  OK  ]
  mysql         [  OK  ]
  ping          [  OK  ]
  status        [  OK  ]

bcm-head-02 -> bcm-head-01*
  failoverping  [  OK  ]
  mysql         [  OK  ]
  ping          [  OK  ]
  status        [  OK  ]
```

44. Power on the cluster nodes.

```
# cmsh -c "power -c dgx on"
ipmi0 ..................... [   ON    ] dgx01
ipmi0 ..................... [   ON    ] dgx02
ipmi0 ..................... [   ON    ] dgx03
ipmi0 ..................... [   ON    ] dgx04
```

45. Configure the Jupyter service on the head node by running `/opt/bcm/provisioning/install_jupyter`.

```
% device
% use bcm-head-02
% services
% use cm-jupyterhub
```

46. Set the `runif` parameter to `ACTIVE`.

```
% set runif active
% commit

% show
Parameter                      Value
------------------------------ ------------------------------------------
Revision
Service                        cm-jupyterhub
Run if                         ACTIVE
Monitored                      yes
Autostart                      yes
Timeout                        -1
Belongs to role                yes
Sickness check script
Sickness check script timeout  10
Sickness check interval        60
```

47. Install Slurm.

    Slurm is installed by running `/opt/bcm/provisioning/install_slurm` and takes place in two parts.

48. Reboot all the non-headnode systems involved with Slurm.

```
cmsh
device
reboot -c slogin
reboot -c dgxnodes
```

49. Modify the `slurmclient-gpu` role to remove the `slurm-client role` and convert `slurm-client-gpu` to use that name instead to simplify the configuration.

```
cmsh
configurationoverlay
remove slurm-client
commit
use slurm-client-gpu
set name slurm-client
commit
roles
use slurmclient
```

50. Clear the `Type` value and set the correct core association with each GPU entry for maximum performance.

```
genericresources
use gpu0
clear type
set cores 48-63,176-191
use gpu1
clear type
set cores 48-63,176-191
use gpu2
clear type
set cores 16-31,144-159
use gpu3
clear type
set cores 16-31,144-159
use gpu4
clear type
set cores 112-127,240-255
use gpu5
clear type
set cores 112-127,240-255
use gpu6
clear type
set cores 80-95,210-223
use gpu7
clear type
set cores 80-95,210-223
commit
```

The `gres.conf` file will be updated automatically by Base Command Manager—these settings align with the expectations of various scripts and tools in the NVIDIA ecosystem and will then maximize compatibility of this environment with those scripts and tools.

Notice

This document is provided for information purposes only and shall not be regarded as a warranty of a certain functionality, condition, or quality of a product. NVIDIA Corporation (“NVIDIA”) makes no representations or warranties, expressed or implied, as to the accuracy or completeness of the information contained in this document and assumes no responsibility for any errors contained herein. NVIDIA shall have no liability for the consequences or use of such information or for any infringement of patents or other rights of third parties that may result from its use. This document is not a commitment to develop, release, or deliver any Materials (defined below), code, or functionality.

NVIDIA reserves the right to make corrections, modifications, enhancements, improvements, and any other changes to this document, at any time without notice.

Customer should obtain the latest relevant information before placing orders and should verify that such information is current and complete.

NVIDIA products are sold subject to the NVIDIA standard terms and conditions of sale supplied at the time of order acknowledgement, unless otherwise agreed in an individual sales agreement signed by authorized representatives of NVIDIA and customer (“Terms of Sale”). NVIDIA hereby expressly objects to applying any customer general terms and conditions with regards to the purchase of the NVIDIA product referenced in this document. No contractual obligations are formed either directly or indirectly by this document.

No license, either expressed or implied, is granted under any NVIDIA patent right, copyright, or other NVIDIA intellectual property right under this document. Information published by NVIDIA regarding third-party products or services does not constitute a license from NVIDIA to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property rights of the third party, or a license from NVIDIA under the patents or other intellectual property rights of NVIDIA.

Reproduction of information in this document is permissible only if approved in advance by NVIDIA in writing, reproduced without alteration and in full compliance with all applicable export laws and regulations, and accompanied by all associated conditions, limitations, and notices.

THIS DOCUMENT AND ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, “MATERIALS”) ARE BEING PROVIDED “AS IS.” NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE. TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL NVIDIA BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF NVIDIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Notwithstanding any damages that customer might incur for any reason whatsoever, NVIDIA’s aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms of Sale for the product.

Trademarks

NVIDIA, the NVIDIA logo, DGX, DGX NVIDIA SuperPOD, and NVIDIA Base Command Manager are trademarks and/or registered trademarks of NVIDIA Corporation in the U.S. and other countries. Other company and product names may be trademarks of the respective companies with which they are associated.

Copyright

NVIDIA Corporation | 2788 San Tomas Expressway, Santa Clara, CA 95051
http://www.nvidia.com