

VIPRION[®] Systems: Blade Migration

Version 12.0



Table of Contents

Legal Notices.....	5
Legal notices.....	5
Migrating the Configuration on B2000 Series Blades.....	7
Overview: Configuration migration from a B2100 blade to a B2150/B2250 blade.....	7
About prerequisites for running the chassis upgrade script.....	7
About sample configuration migration tasks.....	7
Example 1: Replacing multiple blades.....	8
Example 2: Replacing a single blade.....	9
Migration result.....	10
Chassis upgrade command reference.....	10
Migrating vCMP Virtual Disks to B2000 Series Blades.....	11
Overview: Virtual disk migration on B2000 Series blades.....	11
About prerequisites for running the virtual disk migration script.....	11
About sample virtual disk migration tasks.....	12
Example 1: Replacing two blades.....	12
Example 2: Replacing four blades.....	15
Example 3: Consolidating four blades to two blades.....	17
Example 4: Consolidating three blades to two blades.....	19
Example 5: Replacing a single blade.....	22
Example 6: Migrating large virtual disks.....	24
Migration result.....	26
Chassis virtual disk migration command reference.....	27

Legal Notices

Legal notices

Publication Date

This document was published on March 8, 2016.

Publication Number

MAN-0492-02

Copyright

Copyright © 2016, F5 Networks, Inc. All rights reserved.

F5 Networks, Inc. (F5) believes the information it furnishes to be accurate and reliable. However, F5 assumes no responsibility for the use of this information, nor any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent, copyright, or other intellectual property right of F5 except as specifically described by applicable user licenses. F5 reserves the right to change specifications at any time without notice.

Trademarks

AAM, Access Policy Manager, Advanced Client Authentication, Advanced Firewall Manager, Advanced Routing, AFM, APM, Application Acceleration Manager, Application Security Manager, AskF5, ASM, BIG-IP, BIG-IP EDGE GATEWAY, iWorkflow, Cloud Extender, Cloud Manager, CloudFucious, Clustered Multiprocessing, CMP, COHESION, Data Manager, DDoS Frontline, DDoS SWAT, Defense.Net, defense.net [DESIGN], DevCentral, DevCentral [DESIGN], DNS Express, DSC, DSI, Edge Client, Edge Gateway, Edge Mobile, Edge Mobility, Edge Portal, ELEVATE, EM, ENGAGE, Enterprise Manager, F5, F5 [DESIGN], F5 Agility, F5 Certified [DESIGN], F5 Networks, F5 SalesXchange [DESIGN], F5 Synthesis, f5 Synthesis, F5 Synthesis [DESIGN], F5 TechXchange [DESIGN], Fast Application Proxy, Fast Cache, FCINCO, Global Traffic Manager, GTM, GUARDIAN, iApps, IBR, iCall, iControl, iHealth, Intelligent Browser Referencing, Intelligent Compression, IPv6 Gateway, iQuery, iRules, iRules OnDemand, iSession, L7 Rate Shaping, LC, Link Controller, LineRate, LineRate Point, LineRate Precision, LineRate Systems [DESIGN], Local Traffic Manager, LROS, LTM, Message Security Manager, MobileSafe, MSM, OneConnect, Packet Velocity, PEM, Policy Enforcement Manager, Protocol Security Manager, PSM, Ready Defense, Real Traffic Policy Builder, SalesXchange, ScaleN, SDAS (except in Japan), SDC, Signalling Delivery Controller, Solutions for an application world, Software Designed Application Services, Silverline, SSL Acceleration, SSL Everywhere, StrongBox, SuperVIP, SYN Check, SYNTHESIS, TCP Express, TDR, TechXchange, TMOS, TotALL, TDR, TMOS, Traffic Management Operating System, Traffix, Traffix [DESIGN], Transparent Data Reduction, UNITY, VAULT, vCMP, VE F5 [DESIGN], Versafe, Versafe [DESIGN], VIPRION, Virtual Clustered Multiprocessing, WebSafe, and ZoneRunner, are trademarks or service marks of F5 Networks, Inc., in the U.S. and other countries, and may not be used without F5's express written consent.

All other product and company names herein may be trademarks of their respective owners.

Patents

This product may be protected by one or more patents indicated at: <https://f5.com/about-us/policies/patents>

Migrating the Configuration on B2000 Series Blades

Overview: Configuration migration from a B2100 blade to a B2150/B2250 blade

If you are using VIPRION[®] B2100 (original) blades in your VIPRION 2000 Series chassis and would like to upgrade to B2150 or B2250 (replacement) blades, you can use a chassis upgrade script to pull the user configuration set (UCS) from an old blade, and then copy and load the configuration to a new blade. The chassis upgrade script is automated, but requires user input.

To view help options for the script, type: `chassisConfigUpgrade.pm --help`.

Important: During this process, blades should not be processing production traffic and should not be in an Offline state.

Important: If you are upgrading to B2250 blades, you must reassign the new interfaces to the appropriate VLANs.

About prerequisites for running the chassis upgrade script

Before you run the chassis upgrade script, check for these conditions:

Condition	Action
Chassis must not be in a production environment.	Divert your application traffic to another chassis.
Original blade must be running BIG-IP [®] version 11.4.1 HF1 or later.	If the original blade is running an earlier version, you must upgrade the blade software.
vCMP [®] guests on original blades should not be in a Deployed or Provisioned state.	You must put the guests into the Configured state. <i>Note:</i> If vCMP is provisioned on your system, read the section, <i>Migrating vCMP Virtual Disks on B2000 Series Blades</i> .
License must not be expired.	You must activate the license before you can upgrade your VIPRION [®] system.

About sample configuration migration tasks

These examples illustrate how to migrate a configuration from a VIPRION[®] B2100 (original) blade to a B2150/B2250 (new) blade using an automated script. The script requires user input, so you must run it from the serial console of the new blade.

Example 1: Replacing multiple blades

Before you run the script, verify that at least one of the original blades in the chassis is active and running as the primary.

The `chassisConfigUpgrade.pm` script pulls the configuration (UCS) from a B2100 (original) blade, and then copies and loads the configuration to a B2150/B2250 (new) blade.

Important: *You must run the chassis upgrade script from the serial console of the new blade.*

1. Halt all blades except for the one that is installed in slot 2:

- a) Connect to the blade using the serial console.
- b) Halt the blade.

```
slot<N> halt
```

The blade in slot 2 becomes primary.

2. After the Primary LED on slot 2 is lit, insert a new blade into slot 1.
3. Attach a serial cable to the CONSOLE port on the new blade in slot 1.
4. Connect to the serial console of the new blade in slot 1.
5. Log in to the command-line interface of the system using the root account.
6. Run the chassis upgrade script on the new blade.

```
chassisConfigUpgrade.pm
```

Note: *Press Ctrl+C at any time to stop the script.*

7. Monitor the script's progress. The script pauses and displays a message if it encounters any error conditions.

The script displays an inventory of the blades in the cluster, similar to this example:

```
#####  
#STEP 4: cluster member inventory  
#####  
Info: found 2 slots:  
slot 1:  
  primary: 0  
  alive: 1  
slot 2:  
  primary: 1  
  alive: 1
```

8. When prompted, type the root password.
9. Monitor the script's progress. The script stops running and displays a message if it encounters any error conditions.
10. If there are other secondary blades in the cluster, the script prompts you to confirm that you want to stop the blades. If you decline to take down the blades, the script stops running.
11. When prompted for the UCS file name, you can choose either to use the automatically-generated file name, which contains the chassis serial number, or provide your own name.
The script saves the UCS configuration on the primary (original) blade.
12. When prompted for the UCS passphrase, type the preferred passphrase and press Enter. By default, no passphrase is used.
13. Monitor the script's progress. Next, it copies the UCS file from the primary blade and puts it in the `/var/local/ucs` directory on the new blade.

14. When prompted to stop the primary blade, confirm that you want the script to stop the blade. If you decline to stop the primary blade, the script stops running.
The new blade becomes the primary and attempts to load the UCS file. An error message displays if the load fails. For example, if you run out of free space on the new blade. If this occurs, free up space on the blade, and then load the UCS again.
On successful completion, the following message displays: `ChassisConfigUpgrade.pm script completed successfully.`
15. If you have vCMP® provisioned on your system and are running BIG-IP® version 11.4.1 HF1 or later, continue to *Migrating vCMP Virtual Disks to B2000 Series Blades*. Otherwise, continue to the next step.
16. Replace the original blade in slot 2 with a new blade and install any additional new blades.

Note: After you install a blade, wait approximately one to two minutes before installing another to ensure that each blade has sufficient time to boot. When the Status LED is green, the blade is fully booted.

Example 2: Replacing a single blade

Before you run the script, verify that the original blade in the chassis is active and running as the primary. The `chassisConfigUpgrade.pm` script pulls the configuration (UCS) from a B2100 (original) blade, and then copies and loads the configuration to a B2150/B2250 (new) blade.

Important: You must run the chassis upgrade script from the serial console of the new blade.

1. Insert a new blade into slot 2.
2. Attach a serial cable to the CONSOLE port on the new blade in slot 2.
3. Connect to the serial console of the new blade in slot 2.
4. Log in to the command-line interface of the system using the root account.
5. Run the chassis upgrade script on the new blade.

```
chassisConfigUpgrade.pm
```

Note: Press `Ctrl+C` at any time to stop the script.

6. Monitor the script's progress. The script pauses and displays a message if it encounters any error conditions.

The script displays an inventory of the blades in the cluster, similar to this example:

```
#####
#STEP 4: cluster member inventory
#####
Info: found 2 slots:
slot 1:
  primary: 1
  alive: 1
slot 2:
  primary: 0
  alive: 1
```

7. When prompted, type the root password.
8. Monitor the script's progress. The script stops running and displays a message if it encounters any error conditions.

9. When prompted for the UCS file name, you can choose either to use the automatically-generated file name, which contains the chassis serial number, or provide your own name.
The script saves the UCS configuration on the primary (original) blade.
10. When prompted for the UCS passphrase, type the preferred passphrase and press Enter. By default, no passphrase is used.
11. Monitor the script's progress. Next, it copies the UCS file from the primary blade and puts it in the `/var/local/ucs` directory on the new blade.
12. When prompted to stop the primary blade, confirm that you want the script to stop the blade. If you decline to stop the primary blade, the script stops running.
The new blade becomes the primary and attempts to load the UCS file. An error message displays if the load fails. For example, if you run out of free space on the new blade. If this occurs, free up space on the blade, and then load the UCS again.
On successful completion, the following message displays: `ChassisConfigUpgrade.pm script completed successfully.`
13. If you have vCMP[®] provisioned on your system and are running BIG-IP[®] version 11.4.1 HF1 or later, continue to *Migrating vCMP Virtual Disks to B2000 Series Blades*. Otherwise, continue to the next step.
14. (Optional) Install any additional new blades.

Note: After you install a blade, wait approximately one to two minutes before installing another to ensure that each blade has sufficient time to boot. When the Status LED is green, the blade is fully booted.

Migration result

Your original blade configuration is now running on your new blade (which is now the primary blade in the system), and on any additional blades that you installed in the system.

After successful completion of the migration, you will need to delete any volumes with software versions older than BIG-IP 11.4.1 installed on them.

Chassis upgrade command reference

The chassis upgrade script includes these options.

Option	Description
<code>--no-op</code>	Does not run the script, but displays what would have been done.
<code>--verbose</code>	Enables debug logging.
<code>--help</code>	Displays all options.

Migrating vCMP Virtual Disks to B2000 Series Blades

Overview: Virtual disk migration on B2000 Series blades

If you have a VIPRION[®] system that is provisioned for the virtual clustered multi-processing (vCMP[®]) feature, and you are replacing B2100 blades with new B2150 or B2250 blades, you can migrate the virtual disks for the guests to the new blades. You do this after you have run the script `chassisConfigUpgrade` to copy the vCMP host configuration, in the form of a user configuration set (UCS), from an original blade to a new blade.

You can migrate guest configurations to the new blades in either of these ways:

- Run a data collection script named `chassisVdiskMigrate.pm` (recommended method). This script first collects (copies) all virtual disks from the original blades onto the new primary blade. When you subsequently install the remaining new blades, the `chassisVdiskMigrate.pm` script pushes the virtual disks back onto their respective new blades. When using this script, you can verify that all virtual disks were successfully collected from the original blades and pushed to the new blades.
- Create and export a UCS for each guest's BIG-IP[®] configuration, as well as a UCS for the vCMP host configuration. The host configuration contains the definitions of all guests. After installing the new blades, you will need to recreate the guests and import the relevant guest UCS back to each guest. If the guests were running an older version of the BIG-IP software than the BIG-IP version on the new blades, you will need to upload appropriate ISOs and, if necessary, update each guest configuration to specify an initial image.

This section describes how to use the `chassisVdiskMigrate.pm` script to migrate virtual disks to new blades. To assist you in this process, the section provides several examples of migration tasks, where each task differs in the number of original blades and new blades.

You can view help options for the `chassisVdiskMigrate.pm` script at any time by typing the command `chassisVdiskMigrate.pm --help`.

Important: Do not attempt virtual disk migration on a chassis that is currently in a production environment.

About prerequisites for running the virtual disk migration script

Before you begin the virtual disk migration, you should take the following actions.

Action	Explanation
Verify that you have run the <code>chassisConfigUpgrade.pm</code> script.	The blade on which this script was run should now be the primary, active blade. This blade should have the same vCMP [®] host and guest configuration as that of the original cluster.
Verify that the primary, active blade is running BIG-IP [®] version 11.4.1 HF1 or later.	You can do this at the system prompt by typing <code>watch tmsh show sys software status</code> .

Action	Explanation
Label all original blades in the chassis as well as the new replacement blades.	These labels are physical notes that you attach to each blade (original and new) to indicate the blade's current or target slot, as well as the blade serial number.
Verify that you have access to the advanced (bash) shell.	If you do not have access, a system administrator must modify your user account to grant you access to the advanced shell.
Verify that the vCMP feature is provisioned.	You can do this at the system prompt by typing <code>tmsh show sys provision</code> .
Verify that all guests are in the Configured state.	You can do this at the system prompt by typing <code>tmsh show vcmp guest status</code> .
Ensure that the new blade on which you will run the <code>chassisVdiskMigrate.pm</code> script has enough disk space to store all collected virtual disks.	You can do this at the system prompt by typing <code>tmsh show sys disk application-volume</code> .
Verify that the chassis from which you will run the <code>chassisVdiskMigrate.pm</code> script is not in a production environment.	vCMP guests cannot process production traffic while the script is collecting or pushing virtual disks.
Allow adequate time for the <code>chassisVdiskMigrate.pm</code> script to run.	For each virtual disk on a blade, you should expect to wait a minimum of 15 minutes for the <code>collect</code> operation and 15 minutes for the <code>push</code> operation.

About sample virtual disk migration tasks

These examples illustrate multiple ways to migrate a virtual disk migration. You can use the example that most closely matches your VIPRION[®] cluster configuration.

Example 1: Replacing two blades

Before using this sample task, note that the task is based on these conditions:

- Slot 1 contains a new blade and is the primary slot in the cluster.
- Slot 2 contains an original blade.
- Slot 3 contains the original blade taken from slot 1.

When the VIPRION[®] system is provisioned for vCMP[®] and you need to migrate virtual disks from two B2100 blades to two new B2150 or B2250 blades, you can use this sample task. To perform this task, you use the `chassisVdiskMigrate.pm` script.

1. Verify that you are connected to slot 1 through a serial console to the advanced (bash) shell.

Important: *To minimize the chance that an interruption in network connectivity will preempt virtual disk migration, always perform these steps using the serial console of the blade instead of the Secure Shell (SSH) protocol.*

2. Collect all virtual disks from slot 3:

```
chassisVdiskMigrate.pm collect --pull-from slot3 --destined-for slot1
```

This command initially reports the virtual disks on the blade in slot 3 that the system will collect and then asks for confirmation. For example:

```
#####
#STEP 5: Confirm list of vdisks to collect
#####

#####
The following vdisks will be collected:
#####
      test-guest1.img      (20.64 GiB)
      test-guest3.img      (3.89 GiB)
Continue? (y/n) y
```

3. When prompted to continue, type `y`.

The script continues, copying the virtual disks from the original blade in slot 3 to the new blade in slot 1 for later migration to slot 1.

```
#####
#STEP 6: Pulling vdisk test-guest1.img from slot 3
#####

#####
# Successfully pulled vdisk test-guest1.img.
#####

#####
#STEP 7: Pulling vdisk test-guest3.img from slot 3
#####

#####
# Successfully pulled vdisk test-guest3.img.
#####

#####
# Successfully collected 2 vdisks from slot 3!
#####
```

4. Collect all virtual disks from slot 2.

```
chassisVdiskMigrate.pm collect --pull-from slot2 --destined-for slot2
```

This command initially reports the virtual disks on the blade in slot 2 that the system will collect and then asks for confirmation.

5. When prompted to continue, type `y`.

The script continues, copying the virtual disks from the original blade in slot 2 to the new blade in slot 1, for later migration to slot 2.

6. Remove the original blades from slots 2 and 3.

7. Insert the new blade into slot 2 and wait a few minutes.

This causes the system to initialize the new blade, including copying the host configuration from the new primary blade in slot 1 to the other new blade and provisioning the vCMP feature. During this process, one or more reboots occur.

8. Verify that the vCMP feature is provisioned.

```
tmsh show sys provision
```

9. Push the collected virtual disks to the new slots.

```
chassisVdiskMigrate.pm push
```

The script initially shows a list of all collected virtual disks that the script will push to the new blades.

```
#####
#STEP 5: Confirm list of vdisks to push
#####

#####
The following vdisks will be pushed:
#####
    test-guest1.img      (20.64 GiB)
    test-guest3.img      (3.89 GiB)
    test-guest2.img      (2.53 GiB)
    test-guest3.img      (2.24 GiB)
Continue? (y/n) y
```

10. Type y.

The script pushes the collected virtual disks from the new blade in slot 1 to the blades in slots 1 and 2.

```
#####
#STEP 7: Push vdisk test-guest1.img to slot 1
#####

#####
#STEP 8: Push vdisk test-guest3.img to slot 1
#####

#####
#Step 9: Push vdisk test-guest2.img to slot 2
#####

#####
#Step 10: Push vdisk test-guest3.img to slot 2
#####
This may take several minutes; please be patient.
```

11. Verify that all virtual disks were migrated successfully.

```
chassisVdiskMigrate.pm list
```

You should see this information:

```
#####
#          UNCOLLECTED VDISKS          #
#####
                No uncollected vdisks.

#####
#          UNPUSHED VDISKS             #
#####
                No unpushed vdisks.
```

12. Set one or more guests to the Deployed state.

```
tmsh modify vcmp guest all state deployed
```

This step is optional.

13. Delete all virtual disks and metadata created by the collect and push operations.

```
chassisVdiskMigrate.pm delete-and-reset-all
```

This command deletes unnecessary .img files that can consume disk space on the primary blade.

After you perform this task, the chassis is populated with two new blades. Both new blades contain the BIG-IP® configuration data from the original blades, as well as any virtual disks for the relevant vCMP guests.

Example 2: Replacing four blades

Before using this sample task, note that the task is based on these conditions:

- Slot 1 contains a new blade and is the primary slot in the cluster.
- Slots 2, 3, and 4 contain original blades.

When the VIPRION® system is provisioned for vCMP® and you need to migrate virtual disks from four B2100 blades in the chassis to four new B2150 or B2250 blades, you can use this sample task. To perform this task, you use the `chassisVdiskMigrate.pm` script.

1. Verify that you are connected to slot 1 through a serial console to the advanced (`bash`) shell.

Important: To minimize the chance that an interruption in network connectivity will preempt virtual disk migration, always perform these steps using the serial console of the blade instead of the Secure Shell (SSH) protocol.

2. Collect all virtual disks from slot 2.

```
chassisVdiskMigrate.pm collect --pull-from slot2 --destined-for slot2
```

The script initially reports the virtual disks on the blade in slot 2 that the system will collect and then asks for confirmation. For example:

```
#####
#STEP 5: Confirm list of vdisks to collect
#####

#####
The following vdisks will be collected:
#####
      test-guest1.img      (20.64 GiB)
      test-guest3.img      (3.89 GiB)
Continue? (y/n) y
```

3. When prompted to continue, type `y`.

The script continues, copying the virtual disks from the original blade in slot 2 to the new blade in slot 1 for later migration to slot 2.

```
#####
#STEP 6: Pulling vdisk test-guest1.img from slot 2
#####

#####
# Successfully pulled vdisk test-guest1.img.
#####

#####
#STEP 7: Pulling vdisk test-guest3.img from slot 2
#####

#####
# Successfully pulled vdisk test-guest3.img.
#####

#####
```

```
# Successfully collected 2 vdisks from slot 2!
#####
```

4. Collect all virtual disks from slot 3.
`chassisVdiskMigrate.pm collect --pull-from slot3 --destined-for slot3`
 The script initially reports the virtual disks on the blade in slot 3 that the system will collect and then asks for confirmation.
5. When prompted to continue, type `y`.
 The script continues, copying the virtual disks from the original blade in slot 3 to the new blade in slot 1, for later migration to slot 3.
6. Collect all virtual disks from slot 4.
`chassisVdiskMigrate.pm collect --pull-from slot4 --destined-for slot4`
 The script initially reports the virtual disks on the blade in slot 4 that the system will collect and then asks for confirmation.
7. When prompted to continue, type `y`.
 The script continues, copying the virtual disks from the original blade in slot 4 to the new blade in slot 1, for later migration to slot 4.
8. Remove the original blades from slots 2, 3, and 4.
9. Insert the original blade from slot 1 into slot 2 and wait one or two minutes.
 The blade powers up. When this process is finished, you can proceed to the next step.
10. Collect all virtual disks from slot 2.
`chassisVdiskMigrate.pm collect --pull-from slot2 --destined-for slot1`
 The script initially reports the virtual disks on the blade in slot 2 that the system will collect and then asks for confirmation.
11. When prompted to continue, type `y`.
 The script continues, copying the virtual disks from the original blade in slot 2 to the new blade in slot 1, for later migration to slot 1.
12. Remove the original blade in slot 2.
13. Insert the remaining new blades into slots 2, 3, and 4 and wait a few minutes per blade.
 This causes the system to initialize each new blade, including copying the host configuration from the new primary blade in slot 1 to the other new blades and provisioning the vCMP feature. During this process, one or more reboots occur.
14. Verify that the vCMP feature is provisioned.
`tmsh show sys provision`
15. Push the collected virtual disks to the new slots.
`chassisVdiskMigrate.pm push`
 This command initially lists all of the collected virtual disks that the script will push to the new blades.
16. When prompted to continue, type `y`.
 The script pushes the collected virtual disks from the new blade in slot 1 to the new blades.
17. Verify that all virtual disks were migrated successfully.
`chassisVdiskMigrate.pm list`

You should see this information:

```
#####
#          UNCOLLECTED VDISKS          #
#####
          No uncollected vdisks.

#####
#          UNPUSHED VDISKS          #
#####
```



```
No unpushed vdisks.
```

18. Set one or more guests to the Deployed state.

```
tmsh modify vcmp guest all state deployed
```

This step is optional.

19. Delete all virtual disks and metadata created by the `collect` and `push` operations.

```
chassisVdiskMigrate.pm delete-and-reset-all
```

This command deletes unnecessary `.img` files that can consume disk space on the primary blade.

After you perform this task, the chassis is populated with four new blades. All new blades contain the BIG-IP® configuration data from the original blades, as well as any virtual disks for the relevant vCMP guests.

Example 3: Consolidating four blades to two blades

Before using this sample task, note that the task is based on these conditions:

- Slot 1 contains a new blade and is the primary slot in the cluster.
- Slots 2, 3, and 4 contain original blades.

When the VIPRION® system is provisioned for vCMP® and you need to migrate virtual disks from four B2100 blades to two new B2150 or B2250 blades, you can use this sample task. To perform this task, you use the `chassisVdiskMigrate.pm` script.

1. Verify that you are connected to slot 1 through a serial console to the advanced (`bash`) shell.

Important: To minimize the chance that an interruption in network connectivity will preempt virtual disk migration, always perform these steps using the serial console of the blade instead of the Secure Shell (SSH) protocol.

2. Collect all virtual disks from slot 2.

```
chassisVdiskMigrate.pm collect --pull-from slot2 --destined-for slot2
```

The script initially reports the virtual disks on the blade in slot 2 that the system will collect and then asks for confirmation. For example:

```
#####
#STEP 5: Confirm list of vdisks to collect
#####

#####
The following vdisks will be collected:
#####
    test-guest1.img      (20.64 GiB)
    test-guest3.img      (3.89 GiB)
Continue? (y/n) y
```

3. When prompted to continue, type `y`.

The script continues, copying the virtual disks from the original blade in slot 2 to the new blade in slot 1 for later migration to slot 2.

```
#####
#STEP 6: Pulling vdisk test-guest1.img from slot 2
```

```
#####
#####
# Successfully pulled vdisk test-guest1.img.
#####

#####
#STEP 7: Pulling vdisk test-guest3.img from slot 2
#####

#####
# Successfully pulled vdisk test-guest3.img.
#####

#####
# Successfully collected 2 vdisks from slot 2!
#####
```

4. Collect all virtual disks from slot 3.
`chassisVdiskMigrate.pm collect --pull-from slot3 --destined-for slot1`
 The script initially reports the virtual disks on the blade in slot 3 that the system will collect and then asks for confirmation.
5. When prompted to continue, type `y`.
 The script continues, copying the virtual disks from the original blade in slot 3 to the new blade in slot 1, for later migration to slot 1.
6. Collect all virtual disks from slot 4.
`chassisVdiskMigrate.pm collect --pull-from slot4 --destined-for slot2`
 This command initially reports the virtual disks on the blade in slot 4 that the system will collect and then asks for confirmation.
7. When prompted to continue, type `y`.
 The script continues, copying the virtual disks from the original blade in slot 4 to the new blade in slot 1, for later migration to slot 2.
8. Remove the original blades from slots 2, 3, and 4.
9. Insert the original blade from slot 1 into slot 2.
 The blade powers up. You must wait one or two minutes after inserting the blade. When this process is finished, you can proceed to the next step.
10. Collect all virtual disks from slot 2.
`chassisVdiskMigrate.pm collect --pull-from slot2 --destined-for slot1`
 This command initially reports the virtual disks on the blade in slot 2 that the system will collect and then asks for confirmation.
11. When prompted to continue, type `y`.
 The script continues, copying the virtual disks from the original blade in slot 2 to the new blade in slot 1, for later migration to slot 1.
12. Remove the original blade in slot 2.
13. Insert the remaining new blade into slot 2 and wait a few minutes.
 This causes the system to initialize the new blade, including copying the host configuration from the new primary blade in slot 1 to the other new blade and provisioning the vCMP feature. During this process, one or more reboots occur.
14. Verify that the vCMP feature is provisioned.
`tmsh show sys provision`
15. Push all virtual disks to the new blades.
`chassisVdiskMigrate.pm push`
 The script initially lists all of the collected virtual disks that the script will push to the new blades.
16. When prompted to continue, type `y`.
 The script pushes the collected virtual disks from the new blade in slot 1 to the new blades.

- Verify that all virtual disks were migrated successfully.

```
chassisVdiskMigrate.pm list
```

You should see this information:

```
#####
#          UNCOLLECTED VDISKS          #
#####
          No uncollected vdisks.

#####
#          UNPUSHED VDISKS            #
#####
          No unpushed vdisks.
```

- Set one or more guests to the Deployed state.

```
tmsh modify vcmp guest all state deployed
```

This step is optional.

- Delete all virtual disks and metadata created by the `collect` and `push` operations.

```
chassisVdiskMigrate.pm delete-and-reset-all
```

This command deletes unnecessary `.img` files that can consume disk space on the primary blade.

After you perform this task, the chassis is populated with two new blades. Both blades contain the BIG-IP® configuration data from the original blades, as well as any virtual disks for the relevant vCMP guests.

Optionally, to avoid unexpected results, you can reconfigure the guests to ensure that all slots on which they are allowed to run are populated with blades.

Example 4: Consolidating three blades to two blades

Before using this sample task, note that the task is based on these conditions:

- Slot 1 contains a new blade and is the primary slot in the cluster.
- Slots 2 and 3 contain original blades.
- Slot 4 contains the original blade taken from slot 1.

When the VIPRION® system is provisioned for vCMP® and you need to migrate virtual disks from three B2100 blades to two new B2150 or B2250 blades, you can use this sample task. In this example, the system distributes the virtual disks from slot 3 to multiple new slots (slots 1 and 2). To perform this task, you use the `chassisVdiskMigrate.pm` script.

- Verify that you are connected to slot 1 through a serial console to the advanced (`bash`) shell.

Important: To minimize the chance that an interruption in network connectivity will preempt virtual disk migration, always perform these steps using the serial console of the blade instead of the Secure Shell (SSH) protocol.

- Collect all virtual disks from slot 2.

```
chassisVdiskMigrate.pm collect --pull-from slot2 --destined-for slot2
```

The script initially reports the virtual disks on the blade in slot 2 that the system will collect and then asks for confirmation. For example:

```
#####
#STEP 5: Confirm list of vdisks to collect
```

```
#####

#####
The following vdisks will be collected:
#####
    test-guest1.img      (20.64 GiB)
    test-guest3.img      (3.89 GiB)
Continue? (y/n) y
```

- When prompted to continue, type `y`.
The script continues, copying the virtual disks from the original blade in slot 2 to the new blade in slot 1 for later migration to slot 2.

```
#####
#STEP 6: Pulling vdisk test-guest1.img from slot 2
#####

#####
# Successfully pulled vdisk test-guest1.img.
#####

#####
#STEP 7: Pulling vdisk test-guest3.img from slot 2
#####

#####
# Successfully pulled vdisk test-guest3.img.
#####

#####
# Successfully collected 2 vdisks from slot 2!
#####
```

- Collect all virtual disks from slot 4.
`chassisVdiskMigrate.pm collect --pull-from slot4 --destined-for slot1`
This command initially reports the virtual disks on the blade in slot 4 that the system will collect and then asks for confirmation.
- When prompted to continue, type `y`.
The script continues, copying the virtual disks from the original blade in slot 4 to the new blade in slot 1.
- Collect virtual disk `test-guestA.img` from slot 3.
`chassisVdiskMigrate.pm collect --pull-from slot3 --destined-for slot1 test-guestA.img`
The script initially reports the virtual disk that the system will collect and then asks for confirmation.
- When prompted to continue, type `y`.
The script continues, copying the virtual disk `test-guestA.img` from the original blade in slot 3 to the new blade in slot 1, for later migration to slot 1.
- Collect the virtual disk `test-guestB.img` from slot 3.
`chassisVdiskMigrate.pm collect --pull-from slot3 --destined-for slot2 test-guestB.img`
The script initially reports the virtual disk that the system will collect and then asks for confirmation.
- When prompted to continue, type `y`.
The script continues, copying the virtual disk `test-guestB.img` from the original blade in slot 3 to the new blade in slot 1, for later migration to slot 2.
- Verify that all virtual disks were collected.
`chassisVdiskMigrate.pm list`

You should see this information:

```
#####
#          UNCOLLECTED VDISKS          #
#####
          No uncollected vdisks.

#####
#          UNPUSHED VDISKS            #
#####
          No unpushed vdisks.
```

11. Remove the original blades from slots 2, 3, and 4.
12. Insert the remaining new blade into slot 2 and wait a few minutes.
This causes the system to initialize the new blade, including copying the host configuration from the new primary blade in slot 1 to the other new blade and provisioning the vCMP feature. During this process, one or more reboots occur.
13. Verify that the vCMP feature is provisioned.

```
tmsch show sys provision
```
14. Push all virtual disks to the new blades.

```
chassisVdiskMigrate.pm push
```

The script initially lists all of the collected virtual disks that the script will push to the new blades.
15. When prompted to continue, type `y`.
The script pushes the collected virtual disks from the new blade in slot 1 to the new blades.
16. Verify that all virtual disks were migrated successfully.

```
chassisVdiskMigrate.pm list
```

You should see this information:

```
#####
#          UNCOLLECTED VDISKS          #
#####
          No uncollected vdisks.

#####
#          UNPUSHED VDISKS            #
#####
          No unpushed vdisks.
```

17. Set one or more guests to the Deployed state.

```
tmsch modify vcmp guest all state deployed
```

This step is optional.
18. Delete all virtual disks and metadata created by the `collect` and `push` operations.

```
chassisVdiskMigrate.pm delete-and-reset-all
```

This command deletes unnecessary `.img` files that can consume disk space on the primary blade.

After you perform this task, the chassis is populated with two new blades. All new blades contain the BIG-IP® configuration data from the original blades, as well as any virtual disks for the relevant vCMP guests.

Optionally, to avoid unexpected results, you can reconfigure the guests to ensure that all slots on which they are allowed to run are populated with blades.

Example 5: Replacing a single blade

Before using this sample task, note that the task is based on these conditions:

- Slot 1 contains the original blade.
- Slot 2 contains the new blade and is the primary slot in the cluster.

When the VIPRION® system is provisioned for vCMP® and you are replacing one B2100 blade in the chassis with a new B2150 or B2250 blade, you can use this sample task to migrate all virtual disks from the original blade to the new blade. To perform this task, you use the `chassisVdiskMigrate.pm` script.

1. Verify that you are connected to slot 2 through a serial console to the advanced (Bash) shell.

Important: *To minimize the chance that an interruption in network connectivity will preempt virtual disk migration, always perform these steps using the serial console of the blade instead of the Secure Shell (SSH) protocol.*

2. Collect all virtual disks from slot 1:

```
chassisVdiskMigrate.pm collect --pull-from slot1 --destined-for slot2
```

This command initially reports the virtual disks on the blade in slot 1 that the system will collect and then asks for confirmation. For example:

```
#####
#STEP 5: Confirm list of vdisks to collect
#####

#####
The following vdisks will be collected:
#####
      test-guest1.img      (20.64 GiB)
      test-guest3.img      (3.89 GiB)
Continue? (y/n) y
```

3. When prompted to continue, type `y`.

The script continues, copying the virtual disks from the original blade in slot 1 to the new blade in slot 2.

```
#####
#STEP 6: Pulling vdisk test-guest1.img from slot 1
#####

#####
# Successfully pulled vdisk test-guest1.img.
#####

#####
#STEP 7: Pulling vdisk test-guest3.img from slot 1
#####

#####
# Successfully pulled vdisk test-guest3.img.
#####

#####
# Successfully collected 2 vdisks from slot 1!
#####
```

4. Remove the original blade from slot 1.

5. Verify that the vCMP feature is provisioned.

```
tmsch show sys provision
```

6. Push the collected virtual disks to the new blade in slot 2, that is, the same blade on which the virtual disks are collected.

```
chassisVdiskMigrate.pm push
```

This command initially shows a list of all collected virtual disks that the script will push.

```
#####
#STEP 5: Confirm list of vdisks to push
#####

#####
The following vdisks will be pushed:
#####
      test-guest1.img      (20.64 GiB)
      test-guest3.img      (3.89 GiB)
Continue? (y/n) y
```

7. Type `y`.

The script pushes the collected virtual disks to slot 2.

```
#####
#STEP 7: Push vdisk test-guest1.img to slot 2
#####

#####
#STEP 8: Push vdisk test-guest3.img to slot 2
#####

This may take several minutes; please be patient.
```

8. Verify that all virtual disks were migrated successfully.

```
chassisVdiskMigrate.pm list
```

You should see this information:

```
#####
#          UNCOLLECTED VDISKS          #
#####
                No uncollected vdisks.

#####
#          UNPUSHED VDISKS            #
#####
                No unpushed vdisks.
```

9. If you want the new blade to reside in slot 1, remove the blade from slot 2 and insert it into slot 1.

This step is optional.

10. Set one or more guests to the Deployed state.

```
tmsch modify vcmp guest all state deployed
```

This step is optional.

11. Delete all virtual disks and metadata created by the `collect` and `push` operations.

```
chassisVdiskMigrate.pm delete-and-reset-all
```

This command deletes unnecessary `.img` files that can consume disk space on the primary blade.

After you perform this task, slot 2 is populated with one new blade, and all other slots are empty. The new blade contains the BIG-IP® configuration data from the original blade, as well as any virtual disks for the relevant vCMP guests.

Example 6: Migrating large virtual disks

Before using this sample task, note that the task is based on these conditions:

- Slot 1 contains a new blade and is the primary slot in the cluster.
- Slots 2 and 3 contain original blades.

When the VIPRION® system is provisioned for vCMP® and you need to migrate large virtual disks, you can use this task. This task migrates large virtual disks from three B2100 blades to two new B2150 or B2250 blades. To perform this task, you use the `chassisVdiskMigrate.pm` script.

1. Verify that you are connected to slot 1 through a serial console to the advanced (`bash`) shell.

Important: To minimize the chance that an interruption in network connectivity will preempt virtual disk migration, always perform these steps using the serial console of the blade instead of the Secure Shell (SSH) protocol.

2. Collect all virtual disks from slot 2.

```
chassisVdiskMigrate.pm collect --pull-from slot2 --destined-for slot2
```

The script initially reports the virtual disks on the blade in slot 2 that the system will collect and then asks for confirmation. For example:

```
#####
#STEP 5: Confirm list of vdisks to collect
#####

#####
The following vdisks will be collected:
#####
      test-guest1.img      (20.64 GiB)
      test-guest3.img      (3.89 GiB)
Continue? (y/n) y
```

3. When prompted to continue, type `y`.

The script continues, copying the virtual disks from the original blade in slot 2 to the new blade in slot 1 for later migration to slot 2.

```
#####
#STEP 6: Pulling vdisk test-guest1.img from slot 2
#####

#####
# Successfully pulled vdisk test-guest1.img.
#####

#####
#STEP 7: Pulling vdisk test-guest3.img from slot 2
#####

#####
# Successfully pulled vdisk test-guest3.img.
#####

#####
```



```
# Successfully collected 2 vdisks from slot 2!
#####
```

- Remove the original blade in slot 2.

- Insert a new blade into slot 2 and wait a few minutes.

This causes the system to initialize the new blade, including copying the host configuration from the new primary blade in slot 1 to the other new blade and provisioning the vCMP feature. During this process, one or more reboots occur.

- Push the collected virtual disks to the new slot.

```
chassisVdiskMigrate.pm push
```

The script initially shows a list of all collected virtual disks that the script will push to the new blade.

```
#####
#STEP 5: Confirm list of vdisks to push
#####

#####
The following vdisks will be pushed:
#####
test-guest1.img      (20.64 GiB)
test-guest3.img      (3.89 GiB)
Continue? (y/n) y
```

- Type `y`.

The script pushes the collected virtual disks from the new blade in slot 1 to the blade in slot 2.

```
#####
#STEP 7: Push vdisk test-guest1.img to slot 2
#####

#####
#STEP 8: Push vdisk test-guest3.img to slot 2
#####

This may take several minutes; please be patient.
```

- Delete all collected virtual disks from the blade in slot 1.

```
chassisVdiskMigrate.pm delete-pushed-img-files
```

- Collect all virtual disks from slot 3.

```
chassisVdiskMigrate.pm collect --pull-from slot3 --destined-for slot1
```

The script initially reports the virtual disks on the blade in slot 3 that the system will collect and then asks for confirmation.

- When prompted to continue, type `y`.

The script continues, copying the virtual disks from the original blade in slot 3 to the new blade in slot 1, for later migration to slot 1.

- Remove the original blade in slot 3.

- Push the collected virtual disks to slot 1.

```
chassisVdiskMigrate.pm push
```

This command initially lists all of the collected virtual disks that the script will push to slot 1.

- When prompted to continue, type `y`.

The script pushes the collected virtual disks from the new blade in slot 1 to slot 1.

- Delete all collected virtual disks from the blade in slot 1.

```
chassisVdiskMigrate.pm delete-pushed-img-files
```

- Verify that the vCMP feature is provisioned.

```
tmsch show sys provision
```

- Verify that all virtual disks were migrated successfully.

```
chassisVdiskMigrate.pm list
```

You should see this information:

```
#####
#          UNCOLLECTED VDISKS          #
#####
          No uncollected vdisks.

#####
#          UNPUSHED VDISKS            #
#####
          No unpushed vdisks.
```

- Set one or more guests to the Deployed state.

```
tmsch modify vcmp guest all state deployed
```

This step is optional.

- Delete all virtual disks and metadata created by the `collect` and `push` operations.

```
chassisVdiskMigrate.pm delete-and-reset-all
```

This command deletes unnecessary `.img` files that can consume disk space on the primary blade.

After you perform this task, the chassis is populated with two new blades. The blades contain the BIG-IP® configuration data from the original blades, as well as any virtual disks for the relevant vCMP guests.

Optionally, to avoid unexpected results, you can reconfigure the guests to ensure that all slots on which they are allowed to run are populated with blades.

Migration result

After using the `chassisVdisksMigrate.pm` script on a vCMP® system, you have a chassis containing one or more new blades, with no original blades. The new blades contain vCMP host configuration data, as well as the virtual disks for the guests.

You can verify that all virtual disks reside on the new blades by typing the command `tmsch show vcmp`.

This table shows sample output:

Name	Slot ID	OS	Status	Disk use (bytes)
test-guest1.img	1	tmos	ready	19.4G
test-guest2.img	2	tmos	ready	2.7G
test-guest3.img	1	tmos	ready	2.6G
test-guest3.img	2	tmos	ready	2.5G

Note: The reported size of a virtual disk after migration sometimes differs from the original size; this is a normal condition. To ensure virtual disk integrity, the `chassisVdiskMigrate.pm` script performs a checksum on the virtual disk files.

Chassis virtual disk migration command reference

The vCMP® virtual disk migration script includes these actions and options.

Action	Description
collect	Copies virtual disks from the specified blade and stores them on the local blade.
push	Pushes virtual disks from the local blade to the specified destination blade.
list	Shows the status of virtual disks being migrated to new blades in the chassis.
delete-pushed-img-files	Deletes all <code>.img</code> files on the local blade.
delete-and-reset-all	Deletes all <code>.img</code> files and metadata on the local blade.

Option	Description
--no-op	Does not run the script, but displays what would have been done.
--verbose	Enables debug logging.
-h or --help	Displays all options.
--pull-from slotX	Identifies the slot to pull from. This option is valid with the <code>collect</code> action only, and is required.
--destined-for slotX	Identifies the slot to push to. This option is valid with the <code>collect</code> action only. If this option is omitted, the destination is the slot specified with the <code>--pull-from slotX</code> option.

Index

B

- BIG-IP software versions *7, 11*
- blade configuration
 - migrating *11*
- blade configuration migration
 - multiple-blade replacement *8*
 - preparing for *7*
 - results of *10*
 - single-blade replacement *9*
 - tasks for *7*
- blade consolidation *17, 19*
- blade replacement *15, 22*
- blades
 - BIG-IP software version requirement *7*
 - interface assignment on B2250 blade *7*
 - migrating configuration *7*

C

- chassisConfigUpgrade.pm script
 - command reference *10*
 - options *10*
 - running *8–9*
 - UCS creation *7, 11*
- chassisVdiskMigrate.pm script
 - about *11*
 - command reference *27*

D

- disk space
 - assessing *11*
- disk space considerations *24*

G

- guest migration, See virtual disk migration.
- guest state
 - requirements for *11*

H

- help
 - finding *11*
 - options for chassisConfigUpgrade.pm script *7*

M

- migration
 - methods described *11*
 - time estimates *11*

O

- Offline state *7*

P

- prerequisites
 - for chassis upgrade *7*
 - for virtual disk migration *11*

R

- results
 - of blade configuration migration *10*
 - of virtual disk migration *26*

S

- software versions *7, 11*

U

- UCS files
 - for virtual disk migration *11*
 - migrating to a new blade *7*
- upgrade script
 - UCS creation *11*

V

- vCMP provisioning
 - assessing *11*
- virtual disk migration
 - blade consolidation *17, 19*
 - command reference *27*
 - four-blade replacement *15*
 - methods for *11*
 - preparing for *11*
 - results of *26*
 - single-blade replacement *22*
 - tasks for *12*
 - two-blade replacement *12*
 - with large file size *24*
- virtual disks
 - deleting *12*
- virtual disk status
 - viewing *27*

W

- wait times
 - during migration *11*

