vCMP® for Appliance Models: Administration

Version 11.5

Table of Contents

Table of Contents

Legal Notices

Legal Notices

Publication Date

This document was published on October 27, 2017.

Publication Number

MAN-0491-02

Copyright

Copyright © 2013-2016, F5 Networks, Inc. All rights reserved.

Trademarks

Patents

This product may be protected by one or more patents indicated at:
http://www.f5.com/about/guidelines-policies/patents

Acknowledgments

This product includes software developed by Bill Paul.

This product includes software developed by Jonathan Stone.

This product includes software developed by Manuel Bouyer.

This product includes software developed by Paul Richards.

This product includes software developed by the NetBSD Foundation, Inc. and its contributors.

This product includes software developed by the Politecnico di Torino, and its contributors.

This product includes software developed by the Swedish Institute of Computer Science and its contributors.

This product includes software developed by the University of California, Berkeley and its contributors.

This product includes software developed by the Computer Systems Engineering Group at the Lawrence Berkeley Laboratory.

This product includes software developed by Christopher G. Demetriou for the NetBSD Project.

This product includes software developed by Adam Glass.

This product includes software developed by Christian E. Hopps.

This product includes software developed by Dean Huxley.

This product includes software developed by John Kohl.

This product includes software developed by Paul Kranenburg.

This product includes software developed by Terrence R. Lambert.

This product includes software developed by Philip A. Nelson.

This product includes software developed by Herb Peyerl.

This product includes software developed by Jochen Pohl for the NetBSD Project.

This product includes software developed by Chris Provenzano.

This product includes software developed by Theo de Raadt.

This product includes software developed by David Muir Sharnoff.

This product includes software developed by SigmaSoft, Th. Lockert.

This product includes software developed for the NetBSD Project by Jason R. Thorpe.

This product includes software developed by Jason R. Thorpe for And Communications, http://www.and.com.

This product includes software developed for the NetBSD Project by Frank Van der Linden.

This product includes software developed for the NetBSD Project by John M. Vinopal.

This product includes software developed by Christos Zoulas.

This product includes software developed by the University of Vermont and State Agricultural College and Garrett A. Wollman.

This product includes software developed by Balazs Scheidler (bazsi@balabit.hu), which is protected under the GNU Public License.

This product includes software developed by Niels Mueller (nisse@lysator.liu.se), which is protected under the GNU Public License.

In the following statement, *This software* refers to the Mitsumi CD-ROM driver: This software was developed by Holger Veit and Brian Moore for use with 386BSD and similar operating systems. *Similar operating systems* includes mainly non-profit oriented systems for research and education, including but not restricted to NetBSD, FreeBSD, Mach (by CMU).

This product includes software developed by the Apache Group for use in the Apache HTTP server project (http://www.apache.org/).

This product includes software licensed from Richard H. Porter under the GNU Library General Public License (© 1998, Red Hat Software), www.gnu.org/copyleft/lgpl.html.

This product includes the standard version of Perl software licensed under the Perl Artistic License (© 1997, 1998 Tom Christiansen and Nathan Torkington). All rights reserved. You may find the most current standard version of Perl at http://www.perl.com.

This product includes software developed by Jared Minch.

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/).

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com).

This product contains software based on oprofile, which is protected under the GNU Public License.

This product includes RRDtool software developed by Tobi Oetiker (http://www.rrdtool.com/index.html) and licensed under the GNU General Public License.

This product contains software licensed from Dr. Brian Gladman under the GNU General Public License (GPL).

This product includes software developed by the Apache Software Foundation (http://www.apache.org/).

This product includes Hypersonic SQL.

This product contains software developed by the Regents of the University of California, Sun Microsystems, Inc., Scriptics Corporation, and others.

This product includes software developed by the Internet Software Consortium.

This product includes software developed by Nominum, Inc. (http://www.nominum.com).

This product contains software developed by Broadcom Corporation, which is protected under the GNU Public License.

This product contains software developed by MaxMind LLC, and is protected under the GNU Lesser General Public License, as published by the Free Software Foundation.

This product includes Intel QuickAssist kernel module, library, and headers software licensed under the GNU General Public License (GPL).

This product includes software licensed from Gerald Combs (gerald@wireshark.org) under the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or any later version. Copyright ©1998 Gerald Combs.

This product includes software developed by Thomas Williams and Colin Kelley. Copyright ©1986 - 1993, 1998, 2004, 2007

Permission to use, copy, and distribute this software and its documentation for any purpose with or without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation. Permission to modify the software is granted, but not the right to distribute the complete modified source code. Modifications are to be distributed as patches to the released version. Permission to distribute binaries produced by compiling modified sources is granted, provided you

1. distribute the corresponding source modifications from the released version in the form of a patch file along with the binaries,

2. add special version identification to distinguish your version in addition to the base release version number,
3. provide your name and address as the primary contact for the support of your modified version, and
4. retain our contact information in regard to use of the base software.

Permission to distribute the released version of the source code along with corresponding source modifications in the form of a patch file is granted with same provisions 2 through 4 for binary distributions. This software is provided "as is" without express or implied warranty to the extent permitted by applicable law.

This product contains software developed by Google, Inc. Copyright ©2011 Google, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

This product includes software developed by Jeremy Ashkenas and DocumentCloud, and distributed under the MIT license. Copyright © 2010-2013 Jeremy Ashkenas, DocumentCloud.

This product includes gson software, distributed under the Apache License version 2.0. Copyright © 2008-2011 Google Inc.

Introduction to the vCMP System

What is vCMP?

Virtual Clustered Multiprocessing™ *(vCMP*®*)* is a feature of the BIG-IP® system that allows you to provision and manage multiple, hosted instances of the BIG-IP software on a single hardware device. A vCMP hypervisor allocates a dedicated amount of CPU, memory, and storage to each BIG-IP instance. As a vCMP system administrator, you can create BIG-IP instances and then delegate the management of the BIG-IP software within each instance to individual administrators.

A key part of the vCMP system is its built-in flexible resource allocation feature. With flexible resource allocation, you can instruct the hypervisor to allocate a different amount of resource to each BIG-IP instance according to the particular needs of each instance. The hypervisor provides these resources to BIG-IP instances in the form of cores, which contain a portion of system CPU and memory.

At a high level, the vCMP system includes two main components:

vCMP host

> The *vCMP host* is the system-wide hypervisor that makes it possible for you to create and view BIG-IP instances, known as *guests*. Through the vCMP host, you can also manage guest properties. For each guest, the vCMP host allocates system resources according to the particular resource needs of the guest.

vCMP guests

> A *vCMP guest* is an instance of the BIG-IP software that you create on the vCMP system for the purpose of provisioning one or more BIG-IP® modules to process application traffic. A guest consists of a TMOS® instance, plus one or more BIG-IP modules. Each guest has its own share of hardware resources that the vCMP host allocates to the guest, as well as its own management IP addresses, self IP addresses, virtual servers, and so on. This effectively allows each guest to function as a separate BIG-IP device, configured to receive and process application traffic, with no knowledge of other guests on the system. Furthermore, each guest can use TMOS® features such as route domains and administrative partitions to create its own multi-tenant configuration. Each guest requires its own guest administrator to provision, configure, and manage BIG-IP modules within the guest. Depending on the hardware model, a vCMP system can support up to approximately six guests.

This illustration shows a basic vCMP system with a host and four guests. Note that each guest a different set of modules provisioned, depending on the guest's particular traffic requirements.
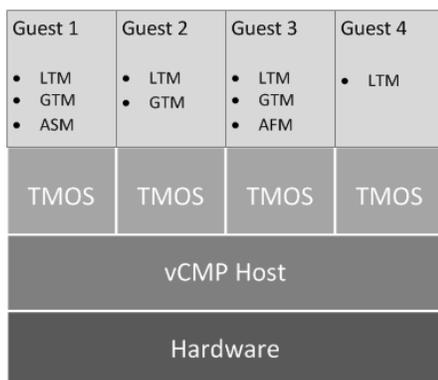
Figure 1: Example of a four-guest vCMP system

Other vCMP system components

In addition to the host and guests, the vCMP® system includes these components:

Virtual disk
A *virtual disk* is an amount of disk space that the system has allocated to a guest (typically 100 gigabytes). Each virtual disk is implemented as an image file with an .img extension, such as guest_A.img.

Core
A *core* is a portion of CPU and memory that the vCMP host allocates to a guest. The amount of CPU and memory that a core provides varies by hardware platform.

BIG-IP license considerations for vCMP

The BIG-IP® system license authorizes you to provision the vCMP® feature and create guests with one or more BIG-IP system modules provisioned. Note the following considerations:

• Each guest inherits the license of the vCMP host.
• The host license must include all BIG-IP modules that are to be provisioned across all guest instances. Examples of BIG-IP modules are BIG-IP Local Traffic Manager™ and BIG-IP Global Traffic Manager™.
• The license allows you to deploy the maximum number of guests that the platform allows.
• If the license includes the appliance mode feature, you cannot enable appliance mode for individual guests; when licensed, appliance mode applies to all guests and cannot be disabled.

You activate the BIG-IP system license when you initially set up the vCMP host.

vCMP provisioning

To enable the vCMP® feature, you perform two levels of provisioning. First, you provision the vCMP feature as a whole. When you do this, the BIG-IP® system, by default, dedicates most of the disk space to running the vCMP feature, and in the process, creates the host portion of the vCMP system. Second, once you have configured the host to create the guests, each guest administrator logs in to the relevant guest and provisions the required BIG-IP modules. In this way, each guest can run a different combination of modules. For example, one guest can run BIG-IP® Local Traffic Manager™ (LTM®) only, while a second guest can run LTM® and BIG-IP ASM™.

Important: Once you provision the vCMP feature, you cannot provision any BIG-IP modules, such as BIG-IP LTM, on the vCMP host. Moreover, if any BIG-IP modules are already provisioned on the system before you provision the vCMP feature, those modules are de-provisioned when you provision the vCMP feature. This, in turn, interrupts any application traffic currently being processed.

Note: The reserved disk space protects against any possible resizing of the file system.

Network isolation

The vCMP® system separates the data plane network from the management network. That is, the host operates with the hardware switch fabric to control the guest data plane traffic. This provides true multi-tenancy by ensuring that traffic for a guest remains separate from all other guest traffic on the system.

The following illustration shows the separation of the data plane network from the management network.
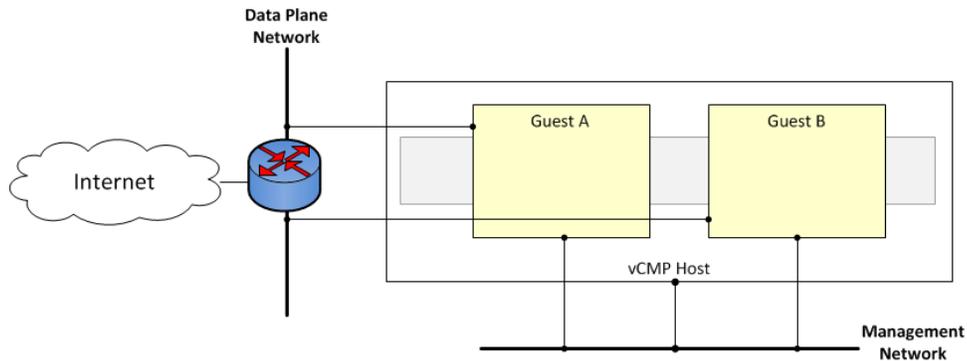
Figure 2: Isolation of the data plane network from the management network

System administration overview

Administering a vCMP® system requires two distinct types of administrators: a vCMP host administrator who creates guests and allocates resources to those guests, and a vCMP guest administrator who provisions and configures BIG-IP ®modules within a specific guest.

At a minimum, these tasks must be performed on the vCMP host, by a host administrator:

* Provision the vCMP feature
* Create vCMP guests, including allocating system resources to each guest
* Create and manage VLANs
* Create and manage trunks
* Manage interfaces
* Configure access control to the host by other host administrators, through user accounts and roles, partition access, and so on

These tasks are performed on a vCMP guest by a guest administrator:

* Provision BIG-IP modules
* Create self IP addresses and associate them with host VLANs
* Create and manage features within BIG-IP modules, such as virtual servers, pools, policies, and so on
* Configure device service clustering (DSC®)
* Configure access control to the guest by other guest administrators, through user accounts and roles, partition access, and so on

Important: vCMP host administration tasks can only be performed on the vCMP host and not from within a guest. This prevents a guest administrator from accessing either the host or other guests on the system, thereby ensuring the separation of administrative tasks across the vCMP deployment.

After you initially set up the vCMP host, you will have a standalone, multi-tenant vCMP system with some number of guests defined. A guest administrator will then be ready to provision and configure the BIG-IP modules within a guest to process application traffic. Optionally, if the host administrator has set up a second system with equivalent guests, a guest administrator can configure high availability for any two equivalent guests.

Guest access to the management network

As a vCMP host administrator, you can configure each vCMP® guest to be either bridged to or isolated from the management network.

Important: F5 Networks recommends that you configure all vCMP guests to be bridged to the management network, unless you have a specific business or security requirement that requires guests to be isolated from the management network, or to be isolated from the management network but remain accessible by way of the host-only interface.

About bridged guests

When you create a vCMP® guest, you can specify that the guest is a bridged guest. A *bridged* guest is one that is connected to the management network. This is the default network state for a vCMP guest. This network state bridges the guest's virtual management interface to the physical management interface of the system.

You typically log in to a bridged guest using its cluster management IP address, and by default, guest administrators with the relevant permissions on their user accounts have access to the bash shell, the BIG-IP® Configuration utility, and the Traffic Management Shell (tmsh). However, if per-guest Appliance mode is enabled on the guest, administrators have access to the BIG-IP Configuration utility and tmsh only.

Although the guest and the host share the host's Ethernet interface, the guest appears as a separate device on the local network, with its own MAC address and IP address.

Note that changing the network state of a guest from isolated to bridged causes the vCMP host to dynamically add the guest's management interface to the bridged management network. This immediately connects all of the guest's VMs to the physical management network.

Important: If you want to easily make TCP connections (for SSH, HTTP, and so on) from either the host or the external network to the guest, or from the guest to the host or external network, you can configure a guest's management port to be on the same IP network as the host's management port, with a gateway identical to the host's management gateway. However, you should carefully consider the security implications of doing so.

About isolated guests

When you create a vCMP® guest, you can specify that the guest is an isolated guest. Unlike a bridged guest, an *isolated* guest is disconnected from the management network. As such, the guest cannot communicate with other guests on the system. Also, because an isolated guest has no management IP address for administrators to use to access the guest, the host administrator, after creating the guest, must use the vconsole utility to log in to the guest and create a self IP address that guest administrators can then use to access the guest.

About Appliance mode

Appliance mode is a BIG-IP system feature that adds a layer of security in two ways:

- By preventing administrators from using the `root` user account.
- By granting administrators access to the Traffic Management Shell (`tmsh`) instead of and the advanced (`bash`) shell.

You can implement Appliance mode in one of two ways:

System-wide through the BIG-IP license
You can implement Appliance mode on a system-wide basis through the BIG-IP® system license. However, this solution might not be ideal for a vCMP® system. When a vCMP system is licensed for Appliance mode, administrators for all guests on the system are subject to Appliance mode restrictions. Also, you cannot disable the Appliance mode feature when it is included in the BIG-IP system license.

On a per-guest basis
Instead of licensing the system for Appliance mode, you can enable or disable the appliance mode feature for each guest individually. By default, per-guest Appliance mode is disabled when you create the guest. After Appliance mode is enabled, you can disable or re-enable this feature on a guest at any time.

Note: If the license for the BIG-IP system includes Appliance mode, the system ignores the per-guest Appliance mode feature and permanently enforces Appliance mode for the vCMP host and all guests on the system.

User access restrictions with Appliance mode

When you enable Appliance mode on a guest, the system enhances security by preventing administrators from accessing the root-level advanced shell (`bash`).

For bridged guests
For a bridged guest with Appliance mode enabled, administrators can access the guest through the guest's management IP address. Administrators for a bridged guest can manage the guest using the BIG-IP® Configuration utility and `tmsh`.

For isolated guests
For an isolated guest with Appliance mode enabled, administrators must access a guest through one of the guest's self IP addresses, configured with appropriate port lockdown values. Administrators for an isolated guest can manage the guest using the BIG-IP Configuration utility and `tmsh`.

Important: When you enable Appliance mode on a guest, any accounts with advanced shell access automatically lose that permission and the permission reverts to `tmsh`. If you disable Appliance mode later, you can re-assign advanced shell access to those accounts.

BIG-IP version restrictions with Appliance mode

If you want to use the BIG-IP® version 11.5 Appliance mode feature on a guest, both the host and the guest must run BIG-IP version 11.5 or later.

Warning: *If you enable Appliance mode on a guest, and a previous version of the BIG-IP software is installed in another boot location, a guest administrator with an Administrator user role can boot to the previous version and obtain advanced shell access.*

Additional Network Considerations

About management network modes for a vCMP guest

As a vCMP host administrator, you can configure each vCMP® guest to operate in one of two modes: Bridged or Isolated. The mode you choose specifies whether the guest is bridged to or isolated from the vCMP host's management network. F5 Networks recommends that you configure all vCMP guests to run in Bridged mode, unless you have a specific business or security requirement that requires Isolated mode.

About the Bridged network mode

Bridged mode is the default network mode for a vCMP guest. This mode bridges the guest's virtual management interface to the physical management port. Typically, you configure a guest's management port to be on the same IP network as the host's management port, with a gateway identical to the host's management gateway. This allows you to make TCP connections (for SSH, HTTP, and so on) easily from either the host or the external network to the guest, or from the guest to the host or external network. Although the guest and the host share the host's Ethernet interface, the guest appears as a separate device on the local network, with its own MAC address and IP address.

Important: If appliance mode is enabled on a guest in Bridged mode, access to the Bash shell is disabled for all administrators of that guest. In this case, only the BIG-IP Configuration utility and the `tmsh` command-line interface are available for managing the guest.

About the Isolated network mode

Isolated mode isolates the guest from the management network. A guest in Isolated mode cannot communicate with other guests on the system. Also, the only way that the vCMP host can access the console of a guest that is in Isolated mode is through the console port (using the `vConsole` utility) or through a self IP address on the guest that allows traffic.

Note: Although a guest in Isolated mode is inaccessible through its management interface, guest administrators can configure self IP addresses, routes, access control, and port lockdown settings appropriate to the environment to achieve their individual connectivity goals.

Important: If appliance mode is enabled on a guest in Isolated mode, access to the Bash shell, the `vConsole` utility, and the BIG-IP Configuration utility is disabled for all administrators of that guest. In this case, only `tmsh` is available for managing the guest.

Other considerations

If the guest is in a Deployed state:

- Setting the network mode from Bridged to Isolated mode causes the vCMP host to disconnect the guest's management interface from its bridged management network, which in turn disconnects the guest from the physical management network.
- Setting the network mode from Isolated to Bridged mode causes the vCMP host to dynamically add the guest's management interface to the bridged management network. This immediately connects the guest to the physical management network.

Changing the network mode while the guest is in the Configured or Provisioned state has no immediate effect.

Network separation of Layer 2 and Layer 3 objects

On a vCMP system, you must configure BIG-IP® Layer 2 objects, such as trunks and VLANs, on the vCMP host and then selectively decide which of these objects you want each guest to inherit. Typically, to ensure that each guest's data plane traffic is securely isolated from other guests, the host administrator creates a separate VLAN for each guest to use. Other objects such as self IP addresses, virtual servers, pools, and profiles are configured on the guest by each guest administrator. With this separation of Layer 2 and Layer 3 objects, application traffic is targeted directly to the relevant guest, further allowing each guest to function as a fully-independent BIG-IP® device.

The following illustration shows the separation of Layer 2 objects from higher-layer objects on the vCMP system:
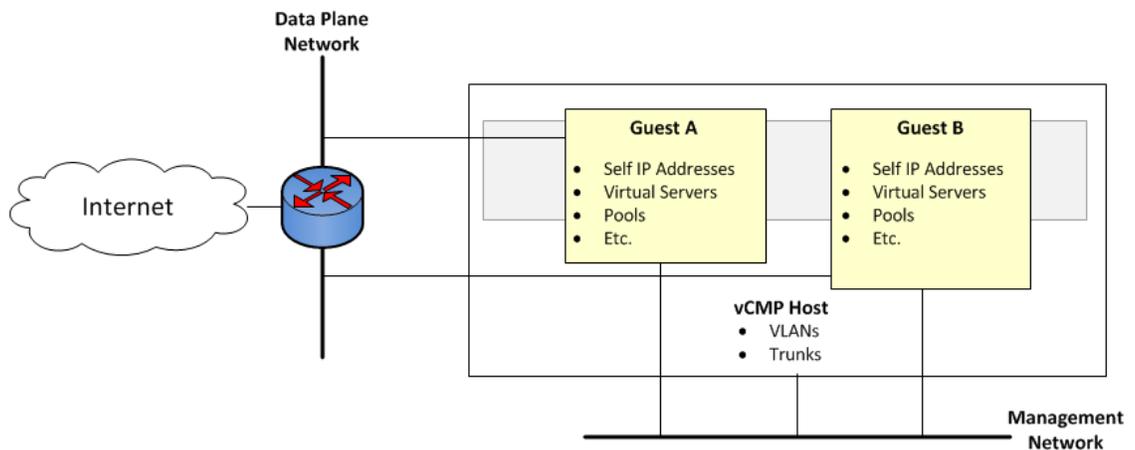
Figure 3: Isolation of network objects on the vCMP system

About the VLAN publishing strategy

For both host and guest administrators, it is important to understand certain concepts about VLAN configuration on a vCMP® system:

• VLAN subscription from host to guest
• System behavior when a host and a guest VLAN have duplicate names or tags

Overview of VLAN subscription

As a vCMP® host administrator, when you create or modify a guest, you typically publish one or more host-based VLANs to the guest. When you *publish* a host-based VLAN to a guest, you are granting a subscription to the guest for use of that VLAN configuration, with the VLAN's underlying Layer 2 resources.

When you publish a VLAN to a guest, if there is no existing VLAN within the guest with the same name or tag as the host-based VLAN, the vCMP system automatically creates, on the guest, a configuration for the published VLAN.

If you modify a guest's properties to remove a VLAN publication from a guest, you are removing the guest's subscription to that host-based VLAN. However, the actual VLAN configuration that the host created within the guest during initial VLAN publication to the guest remains there for the guest to use. In this case, any changes that a host administrator might make to that VLAN are not propagated to the guest.

In general, VLANs that appear within a guest can be either host-based VLANs currently published to the guest, host-based VLANs that were once but are no longer published to the guest, or VLANs that the guest administrator manually created within the guest.

This example shows the effect of publishing a host-based VLAN to, and then deleting the VLAN from, a guest that initially had no VLANs.

```
# Within guest G1, show that the guest has no VLANs configured:

[root@G1:/S1-green-P:Active:Standalone] config # tmsh list net vlan

# From the host, publish VLAN v1024 to guest G1:

[root@host_210:/S1-green-P:Active:Standalone] config # tmsh modify vcmp guest
 G1 vlans add { v1024 }

# Within guest G1, list all VLANs:

[root@G1:/S1-green-P:Active:Standalone] config # tmsh list net vlan

net vlan v1024 {
if-index 96
tag 1024
}

# On the host, delete the host-based VLAN publication from guest G1:

[root@host_210:/S1-green-P:Active:Standalone] config # tmsh modify vcmp guest
 G1 vlans del { v1024 }

# Notice that the host-based VLAN still exists within the guest:

[root@G1:/S1-green-P:Active:Standalone] config # tmsh list net vlan

vlan v1024 {
if-index 96
tag 1024
}
```

About VLANs with identical tags and different names

Sometimes a host administrator might publish a VLAN to a guest, but the guest administrator has already created, or later creates, a VLAN with a different name but the same VLAN tag. In this case, the guest VLAN always overrides the host VLAN. The VLAN can still exist on the host (for other guests to subscribe to), but it is the guest VLAN that is used.

Whenever host and guest VLANs have different names but the same tags, traffic flows successfully across the host from the guest because the VLAN tag alignment is correct. That is, when the tags match, the underlying Layer 2 infrastructure of the VLANs matches, thereby enabling the host to reach the guest.

The example here shows the `tmsh` command sequence for creating two separate VLANs with different names and the same tag, and the resulting successful traffic flow.

```
# On the host, create a VLAN with a unique name but with a tag matching that
 of a guest VLAN VLAN_A:

[root@host_210:/S1-green-P:Active:Standalone] config # tmsh create net vlan
VLAN_B tag 1000

# On the host, publish the host VLAN to the guest:

[root@host_210:/S1-green-P:Active:Standalone] config # tmsh modify vcmp guest
 guest1 vlans add { VLAN_B }

# Within the guest, show that the guest still has its own VLAN only, and not
 the VLAN published from the host:

[root@G1:/S1-green-P:Active:Standalone] config # tmsh list net vlan all

net vlan VLAN_A {
    if-index 192
    tag 1000
    }

# On the guest, create a self IP address for VLAN_A:

[root@G1:/S1-green-P:Active:Standalone] config # tmsh create net self
10.1.1.1/24 vlan VLAN_A

# On the host, delete the self IP address on VLAN_A (this VLAN also exists on
 the guest) and re-create the self IP address on VLAN_B (this VLAN has the
same tag as VLAN_A):

[root@host_210:/S1-green-P:Active:Standalone] config # tmsh delete net self
10.1.1.2/24
[root@host_210:/S1-green-P:Active:Standalone] config # tmsh create net self
10.1.1.2/24 vlan VLAN_B

# From the host, open a connection to the guest, and notice that because the
 two VLANs have the same tags, the connection succeeds:

[root@host_210:/S1-green-P:Active:Standalone] config # ping -c2 10.1.1.1

PING 10.1.1.1 (10.1.1.1) 56(84) bytes of data.
64 bytes from 10.1.1.1: icmp_seq=1 ttl=255 time=3.35 ms
64 bytes from 10.1.1.1: icmp_seq=2 ttl=255 time=0.989 ms

--- 10.1.1.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.989/2.170/3.352/1.182 ms
```

About VLANs with identical names and different tags

Sometimes a host administrator might publish a VLAN to a guest, but the guest administrator has already created, or later creates, a VLAN with the same name but with a different VLAN tag. In this case, the guest VLAN always overrides the host VLAN. The VLAN can still exist on the host (for other guests to subscribe to), but it is the guest VLAN that is used.

Whenever host and guest VLANs have the same names but different tags, traffic cannot flow between the identically-named VLANs at Layer 2. That is, when the tags do not match, the underlying Layer 2 infrastructure of the VLANs does not match, thereby preventing the host from reaching the guest.

The example here shows the `tmsh` command sequence for creating two separate VLANs with the same names and different tags, and the resulting traffic flow issue.

```
# While logged into the guest, create a VLAN:

[root@G1:/S1-green-P:Active:Standalone] config # tmsh create net vlan VLAN_A
 tag 1000

# Show that no VLANs exist on the host:

[root@host_210:/S1-green-P:Active:Standalone] config # tmsh list net vlan all

[root@host_210:/S1-green-P:Active:Standalone] config #

# On the host, create a VLAN with the same name as the guest VLAN but with a
 unique tag on the host:

[root@host_210:/S1-green-P:Active:Standalone] config # tmsh create net vlan
VLAN_A tag 1001

# Publish the host VLAN to the guest:

[root@host_210:/S1-green-P:Active:Standalone] config # tmsh modify vcmp guest
 guest1 vlans add { VLAN_A }

# Within the guest, show that the guest still has its own VLAN only, and not
 the VLAN published from the host:

[root@G1:/S1-green-P:Active:Standalone] config # tmsh list net vlan all

   net vlan VLAN_A {
    if-index 192
    tag 1000
    }

# Within the guest, create a self IP address for the VLAN:

[root@G1:/S1-green-P:Active:Standalone] config # tmsh create net self
10.1.1.1/24 vlan VLAN_A

# On the host, create a self IP address for the identically-named VLAN:

[root@host_210:/S1-green-P:Active:Standalone] config # tmsh create net self
10.1.1.2/24 vlan VLAN_A

# From the host, open a connection to the guest, and notice that because the
 two VLANs have different tags, the connection fails:

[root@host_210:/S1-green-P:Active:Standalone] config # ping -c2 10.1.1.1

PING 10.1.1.1 (10.1.1.1) 56(84) bytes of data.
From 10.1.1.2 icmp_seq=1 Destination Host Unreachable
From 10.1.1.2 icmp_seq=2 Destination Host Unreachable

--- 10.1.1.1 ping statistics ---
2 packets transmitted, 0 received, +2 errors, 100% packet loss, time 3000ms
pipe 2
```

Solution for tag discrepancy between host and guest VLANs

When a host-based VLAN and a guest-created VLAN have identical names but different VLAN tags, traffic flow at Layer 2 is impeded between host and guest. Fortunately, you can resolve this issue by performing these tasks, in the sequence shown:

- Within the guest, delete the relevant VLAN from within the guest.
- On the host, remove the VLAN publication from the guest.
- On the host, modify the tag of the host-based VLAN.
- On the host, publish the VLAN to the guest.
- Within the guest, view the VLAN from within the guest.

Deleting the VLAN within the guest

You use this task when you want to delete a VLAN from within a vCMP® guest. One reason for deleting a VLAN from within a guest is to help resolve a tag discrepancy between a guest VLAN and a host VLAN.

Important: *To perform this task, you must be logged in to the relevant vCMP guest.*

1. On the Main tab, click **Network** > **VLANs**.
   The VLAN List screen opens.
2. In the Name column, locate the name of the VLAN for which you want to change the partition, and to the left of the name, select the check box and click **Delete**.
   The system prompts you to confirm the delete action.
3. Click **Delete**.

After performing this task, you no longer see the VLAN name in the list of VLANs on the guest.

Removing the VLAN publication on the guest

You perform this task when you want to remove a VLAN subscription on a particular guest. One reason for deleting a VLAN from within a guest is to help resolve a tag discrepancy between a guest VLAN and a host VLAN.

Important: *To perform this task, you must be logged in to the vCMP host.*

1. On the Main tab, click **vCMP** > **Guest List**.
   This displays a list of vCMP guests on the system.
2. In the Name column, click the name of the guest that you want to modify.
   This displays the configured properties of the guest.
3. For the **VLAN List** setting, select the relevant VLAN name from the **Selected** list, and use the Move button to move the name to the **Available** list.
4. Click **Update**.

Modifying the tag of the host-based VLAN

You perform this task to change a VLAN tag on a vCMP® host to ensure that the tag matches that of a VLAN on a guest.

Important: *To perform this task, you must be logged in to the vCMP host.*

1. On the Main tab, click **Network** > **VLANs**.
   The VLAN List screen opens.
2. In the Name column, click the relevant VLAN name.
   This displays the properties of the VLAN.
3. In the **Tag** field, type the same tag that was assigned to the VLAN you previously deleted.

4. If the host and guest VLANs have an optional customer tag, type the same customer tag that was assigned to the VLAN you previously deleted.

5. Click **Update**.

Publishing the VLAN to the guest

You perform this task when you want to publish a host-based VLAN to a particular guest.

Important: To perform this task, you must be logged in to the vCMP® host.

1. On the Main tab, click **vCMP** > **Guest List**.
   This displays a list of vCMP guests on the system.

2. In the Name column, click the name of the guest that you want to modify.
   This displays the configured properties of the guest.

3. For the **VLAN List** setting, select the relevant VLAN name from the **Available** list, and use the Move button to move the name to the **Selected** list.

4. Click **Update**.

After performing this task, the guest can use the selected host-based VLAN.

Viewing the new VLAN within the guest

You perform this task to verify that the VLAN that the host published to a guest appears on the guest, with the correct tag.

Important: To perform this task, you must be logged in to the relevant vCMP® guest.

1. On the Main tab, click **Network** > **VLANs**.
   The VLAN List screen opens.

2. In the Name column, click the name of the VLAN that the host published to the guest.

3. In the **Tag** field, verify that the correct tag is shown.

4. Click **Cancel**.

After you perform this task, you can see that the VLAN which the host published to the guest has appeared on the guest, with the correct tag.

Interface assignment for vCMP guests

The virtualized nature of vCMP® guests abstracts many underlying hardware dependencies, which means that there is no direct relationship between guest interfaces and the physical interfaces assigned to VLANs on the vCMP host.

Rather than configuring any interfaces on a guest, a guest administrator simply creates a self IP address within the guest, specifying one of the VLANs that the host administrator previously configured on the host and assigned to the guest during guest creation.

As host administrator, if you want to limit the guest to using specific physical interfaces, you simply change the physical interface assignments on the VLANs that you assign to that guest.

Flexible Resource Allocation

What is flexible resource allocation?

Flexible resource allocation is a built-in vCMP® feature that allows vCMP host administrators to optimize the use of available system resources. Flexible resource allocation gives you the ability to configure the vCMP host to allocate a different amount of CPU and memory to each guest, based on the needs of the specific BIG-IP® modules provisioned within a guest. When you create each guest, you specify the number of cores that you want the host to allocate to the guest. Configuring these settings determines the total amount of CPU and memory that the host allocates to the guest. With flexible allocation, you can customize CPU and memory allocation in granular ways that meet the specific resource needs of each individual guest.

Understanding guest resource requirements

Before you create vCMP® guests and allocate system resources to them, you need to determine the specific CPU and memory needs of each guest. You can then decide how many cores to allocate to a guest, factoring in the resource capacity of your hardware platform.

To determine the CPU and memory resource needs, you must know:

- The number of guests you need to create
- The specific BIG-IP® modules you need to provision within each guest
- The combined memory requirement of all BIG-IP modules within each guest

Resource allocation planning

To determine the resource allocation that you need to configure for each guest, you must know the resource capacity of your hardware platform and the total memory requirement for each guest. You can then calculate the number of cores required for each guest based on the memory requirement that you want to configure for each guest.

Prerequisite hardware considerations

Appliance models vary in terms of how many cores the model provides and how much memory each core contains. Also variable is the maximum number of guests that each model supports.

Before you can determine the number of cores to allocate to a guest, you should understand:

- The total number of cores that the model provides
- The amount of memory that the model provides
- The maximum number of guests that the model supports

By understanding these metrics, you ensure that the total amount of resource you allocate to guests is aligned with the amount of resource that your appliance model supports.

For specific information on the resources that each appliance model provides, see the vCMP® guest memory/CPU core allocation matrix on the AskF5™ Knowledge Base at `http://support.f5.com`.

About core allocation for a guest

When you create a guest on the vCMP® system, you must specify the total number of cores that you want the host to allocate to the guest based on the guest's total resource needs. Each core provides some amount of CPU and a fixed amount of memory. You should therefore specify enough cores to satisfy the combined memory requirements of all BIG-IP® modules that you provision within the guest.

Important: For metrics on memory and CPU support per appliance model, refer to the vCMP guest memory/CPU allocation matrix at `http://support.f5.com`.

About single-core guests

On platforms with hard drives, the vCMP® host always allocates cores for a guest in increments of two cores. In the case of platforms with solid-state drives, however, the host can allocate a single core to a guest, but only for a guest that requires a maximum of one core; for guests that require more than one core, the host does not allocate an odd number of cores (such as three, five, or seven cores).

The illustration shows a possible guest configuration on an appliance with a solid-state drive, where one of the guests has a single core only allocated to it.

Figure 4: A vCMP configuration with a single-core guest

Because the amount of CPU and memory in a single-core guest is limited, F5 Networks® highly recommends that you provision only the BIG-IP® Local Traffic Manager™ (LTM®) module within a single-core guest, and no other modules.

About SSL and compression hardware

On systems that include SSL and compression hardware processors, the vCMP® feature shares these hardware resources among all guests on the system, in a round robin fashion.

When sharing SSL hardware, if all guests are using similar-sized keys, then each guest receives an equal share of the SSL resource. Also, if any guests are not using SSL keys, then other guests can take advantage of the extra SSL resource.

Guest states and resource allocation

As a vCMP® host administrator, you can control when the system allocates or de-allocates system resources to a guest. You can do this at any time, by setting a guest to one of three states: Configured, Provisioned, or Deployed. These states affect resource allocation in these ways:

Configured

This is the initial (and default) state for newly-created guests. In this state, the guest is not running, and no resources are allocated to the guest. If you change a guest from another state to the Configured state, the vCMP host does not delete any virtual disks that were previously attached to that guest; instead, the guest's virtual disks persist on the system. The host does, however, automatically de-allocate other resources such as CPU and memory. When the guest is in the Configured state, you cannot configure the BIG-IP® modules that are licensed to run within the guest; instead, you must set the guest to the Deployed state to provision and configure the BIG-IP modules within the guest.

Provisioned

When you change a guest from Configured to Provisioned, the vCMP host allocates system resources to the guest (CPU, memory, and any unallocated virtual disks). If the guest is new, the host creates new virtual disks for the guest and installs the selected ISO image on them. A guest does not run while in the Provisioned state. When you change a guest from Deployed to Provisioned, the host shuts down the guest but retains its current resource allocation.

Deployed

When you change a guest to the Deployed state, the guest administrator can then provision and configure the BIG-IP modules within the guest. If you are a host administrator and you reconfigure the properties of a guest after its initial deployment, the host immediately propagates those changes to all of the guests and also propagates the list of allowed VLANs.

Device Service Clustering for vCMP Systems

Overview: Device service clustering for vCMP systems

One of the tasks of a vCMP® guest administrator is to configure device service clustering (DSC®). Using *DSC*, a guest administrator can implement config sync, failover, and mirroring across two or more hardware devices. Configuring DSC is the same on a vCMP system as on non-virtualized systems, except that the members of a device group are virtual devices (guests) rather than physical devices.

When configuring DSC, a guest administrator creates a device group that consists of vCMP guests as members, where each member is deployed on a separate BIG-IP device.

For example, a Sync-Failover device group in an active-standby configuration can consist of:

* `guest_A` on `device_1` and `guest_A` on `device_2`
* `guest_B` on `device_1` and `guest_B` on `device_2`
* `guest_C` on `device_1` and `guest_C` on `device_2`

Creating a device group that consists of guests on separate devices ensures that if a device member goes out of service, any active traffic groups on a guest can fail over to another member of the device.

This illustration shows this DSC configuration. The illustration shows two systems, with three guests on each system. Each guest and its equivalent guest on the other system form a separate Sync-Failover device group.
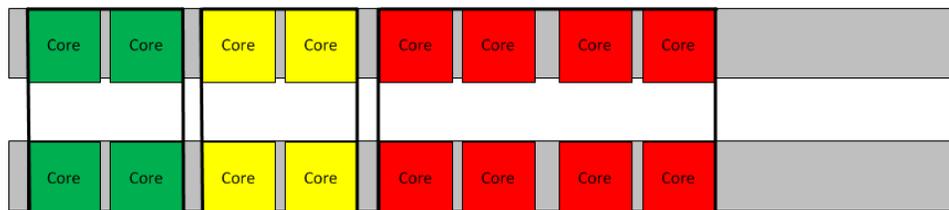
Figure 5: vCMP guests forming three device groups across two appliances

Required IP addresses for DSC configuration

This table describes the types of IP addresses that a guest administrator specifies when configuring device service clustering (DSC®) on a vCMP system.

Table 1: Required IP addresses for DSC configuration on a vCMP system

Configuration feature	IP addresses required
Device trust	The cluster IP address that the vCMP host administrator assigned to the guest during guest creation.
Config sync	Any non-floating self IP address on the guest that is associated with an internal VLAN on the host.

Configuration feature	IP addresses required
Failover	A unicast non-floating self IP address on the guest that is associated with an internal VLAN on the host (preferably VLAN HA) and the management IP address of the device.
Connection mirroring	For both the primary and the secondary IP addresses, a non-floating self IP address on the guest that is associated with an internal VLAN on the host. The secondary address is optional.

Failover methods for vCMP guests

Each traffic group in a device service clustering (DSC®) device group has a property known as a failover method. The *failover method* dictates the way that the system chooses a target device for failover. Available failover methods that the user can choose from are: load-aware failover, an ordered list, and an HA group.

The specific core allocation for a guest in a Sync-Failover device group determines the particular failover method that is appropriate for a DSC traffic group within the guest:

- Guests in a device group that are identical in terms of core allocation are considered to be *homogeneous* guests. In this case, an ordered list would be an appropriate failover method, since relative capacity is equal among all guests.
- Guests in a device group that differ from one another in terms of core allocation are considered to be *heterogeneous* guests. In this case, load-aware failover is an appropriate failover method because the guest administrator can define a relative capacity and relative traffic load for each guest.

An additional type of failover method is an HA group, which applies to both homogeneous and heterogeneous guests.

About HA groups for vCMP systems

For failover configuration, an alternative to using load-aware failover or an ordered list is to use HA groups. An *HA group* is a specification of certain pools or host trunks (or any combination of these) that a guest administrator associates with a traffic group instance. The most common reason to configure an HA group is to ensure that failover is triggered when some number of trunk members become unavailable.

The BIG-IP® system uses an HA group to calculate an overall health score for the instance of a traffic group on a guest. The instance of a traffic group that has the best overall score at any given time becomes or remains the active traffic group instance. With an HA group, the system triggers failover of a traffic group based on changes to trunk or pool health instead of on system, gateway, or VLAN failure.

Because trunks and HA groups are never synchronized among guests as part of a config sync operation, you must assign a separate HA group to each traffic group instance. For example, you could create `ha_group_A` to reference the host trunk `my_trunk` and assign the HA group to `traffic-group-1` on `guest_A`. You could then create another HA group, `ha_group_B`, to also reference `my_trunk` and assign the HA group to the same traffic group (`traffic-group-1`)on `guest_B`.

About connection mirroring for vCMP systems

Connection mirroring is a device service clustering (DSC®) feature that allows a device to mirror its connection and persistence information to another device. Connection mirroring prevents interruption in service during failover. On a vCMP system, the devices that mirror their connections to each other are virtual devices (vCMP guests).

Within any Sync-Failover device group on a vCMP system, connection mirroring can only be implemented between exactly two guests, and the two guests must reside on separate hardware devices and be homogeneous. The term *homogeneous* normally refers to platforms that are the same model of hardware, but in the context of vCMP, the term refers to guests that are equivalent in terms of resource allocation; that is, the guests have the same number of cores allocated to them.

Initial vCMP Configuration Tasks

Overview: vCMP application volume management

The BIG-IP® system allocates all but 30 gigabytes of the total disk space to the vCMP® application volume. Known as the *reserve disk space*, the remaining 30 gigabytes of disk space are left available for other uses, such as for installing additional versions of the BIG-IP system in the future. The vCMP disk space allocation, as well as the creation of the reserve disk space, occurs when you initially provision the vCMP feature as part of vCMP host configuration.

If you want the system to reserve more than the standard 30 gigabytes of disk space for non-vCMP uses, you must do this prior to provisioning the vCMP feature. Adjusting the reserved disk space after you have provisioned the vCMP feature can produce unwanted results.

Important: When increasing the reserve disk space for additional BIG-IP installations, the recommended amount of space to reserve is 8 gigabytes per installation.

Viewing disk space allocation for a vCMP application volume

Using this procedure, you can view the amount of disk space, in megabytes, that the system has allocated to a vCMP application volume.

1. In the URL field, type the management IP address that you previously assigned to the system.

   `https://<ip_address>`

   The browser displays the login screen for the BIG-IP Configuration utility.
2. On the Main tab, click **System** > **Disk Management**.
   The display shows the logical disks and application volumes from the perspective of the vCMP host.
3. Click the logical disk for which you want to reserve disk space.
   An example of a logical disk is `HD1`.
4. On the menu bar, click **Image List** if displayed.

   The screen displays a list of the installed images on the system.
5. If a list of images appears, locate the relevant image, and in the Disk column, click the logical disk name.
6. In the Contained Application Volumes area of the screen, in the Volume column, locate the vCMP application volume.
7. In the Size (MB) column, view the size of the application volume, in megabytes.

Modifying disk space allocation for a vCMP application volume

When you provision the BIG-IP system for vCMP, the BIG-IP system dedicates all but 30 gigabytes of disk space to running the vCMP feature. (The 30 gigabytes of reserved disk space protects against any possible resizing of the file system.) Before provisioning the vCMP feature, you can reserve additional space for a

logical disk. Use this procedure if you decide that you need to change the amount of disk space (in megabytes) that the system allocates to a vCMP application volume.

1. In the URL field, type the management IP address that you previously assigned to the system.

   ```
   https://<ip_address>
   ```

   The browser displays the login screen for the BIG-IP Configuration utility.

2. On the Main tab, click **System** > **Disk Management**.
   The display shows the logical disks and application volumes from the perspective of the vCMP host.

3. Click the logical disk for which you want to reserve disk space.
   An example of a logical disk is HD1.

4. On the menu bar, click **Image List** if displayed.

   The screen displays a list of the installed images on the system.

5. If a list of images appears, locate the relevant image, and in the Disk column, click the logical disk name.

6. In the **Reserved (MB)** field, increase the amount of disk space that you want to reserve for the logical disk.

   The more space your reserve, the less disk space is available for the vCMP application volume.

7. Click **Update**.

vCMP host administrator tasks

As a vCMP® host administrator, you have the important task of initially planning the amount of total system CPU and memory that you want the vCMP host to allocate to each guest. This decision is based on the resource needs of the particular BIG-IP® modules that guest administrators intend to provision within each guest, as well as the maximum system resource limits for the relevant hardware platform. Thoughtful resource allocation planning prior to creating the guests ensures optimal performance of each guest. Once you have determined the resource allocation requirements for the guests, you are ready to configure the host. Overall, your primary duties are to provision the vCMP feature and to create and manage guests, ensuring that the proper system resources are allocated to those guests.

Task summary

Accessing the vCMP host

Performing this task allows you to access the vCMP host. Primary reasons to access the host are to create and manage vCMP® guests, manage virtual disks, and view or manage host and guest properties. You can also view host and guest statistics.

1. From a system on the external network, display a browser window.

2. In the URL field, type the management IP address that you previously assigned to the system, as follows:

   ```
   https://<ip_address>
   ```

   The browser displays the login screen for the BIG-IP® Configuration utility.

Provisioning the vCMP feature

Before performing this task, ensure that the amount of reserve disk space that the provisioning process creates is sufficient. Attempting to adjust the reserve disk space after you have provisioned the vCMP® feature produces unwanted results.

Performing this task creates the vCMP host (the hypervisor) and dedicates most of the system resources to running vCMP.

Warning: If the system currently contains any BIG-IP® module configuration data, this data will be deleted when you provision the vCMP feature.

1. On the Main tab, click **System** > **Resource Provisioning**.
2. Verify that all BIG-IP modules are set to **None**.
3. From the **vCMP** list, select **Dedicated**.
4. Click **Update**.

After provisioning the vCMP feature, the system reboots TMOS® and prompts you to log in again. This action logs you in to the vCMP host, thereby allowing you to create guests and perform other host configuration tasks.

Creating a vCMP guest

Before creating a guest on the system:

- Verify that you have configured the base network on the system to create any necessary trunks, as well as VLANs for guests to use when processing application traffic.
- If you plan to enable the **Appliance Mode** setting for the guest, verify that the vCMP license on the host does not specify appliance mode; if appliance mode is specified in the vCMP license, the feature is applied system-wide to the host and to all guests on the system, instead of on a per-guest basis.

You create a vCMP guest when you want to create an instance of the BIG-IP software for the purpose of running one or more BIG-IP® modules to process application traffic. For example, you can create a guest that runs BIG-IP® Local Traffic Manager™ and BIG-IP® Global Traffic Manager™. When creating a guest, you specify the number of cores that you want the vCMP host to allocate to each guest.

Note: When creating a guest, if you see an error message such as `Insufficient disk space on /shared/vmdisks. Need 24354M additional space.`, *you must delete existing unattached virtual disks until you have freed up that amount of disk space.*

1. Use a browser to log in to system, using the management IP address.
   This logs you in to the vCMP® host.
2. On the Main tab, click **vCMP** > **Guest List**.
   This displays a list of vCMP guests on the system.
3. Click **Create**.
4. From the **Properties** list, select **Advanced**.
5. In the **Name** field, type a name for the guest.
6. In the **Host Name** field, type a fully-qualified domain name (FQDN) name for the guest.
   If you leave this field blank, the system assigns the name `localhost.localdomain`.
7. From the **Cores Per Guest** list, select the number of cores that you want the host to allocate to the guest.

8. From the **Management Network** list, select a value:

   - If you want the guest to be connected to the management network, select the recommended value, **Bridged**.
   - If you do not want the guest to be connected to the management network, select **Isolated**.

   *Important: If you select **Isolated**, do not enable the **Appliance Mode** setting when you initially create the guest. For more information, see the step for enabling the **Appliance Mode** setting.*

   Selecting **Bridged** causes the **IP Address** setting to appear.

9. For the **Management Port** setting, fill in the required information:

   a) In the **IP Address** field, type a unique management IP address that you want to assign to the guest.

      You use this IP address to access the guest when you want to manage the BIG-IP modules running within the guest.

   b) In the **Network Mask** field, type the network mask for the management IP address.

   c) In the **Management Route** field, type a gateway address for the management IP address.

   Important: Assigning an IP address that is on the same network as the host management port has security implications that you should carefully consider.

10. From the **Initial Image** list, select an ISO image file for installing TMOS® software onto the guest's virtual disk.

11. In the **Virtual Disk** list, retain the default value of **None**.

    Note that if an unattached virtual disk file with that default name already exists, the system displays a message, and you must manually attach the virtual disk. You can do this using the `tmsh` command line interface, or use the Configuration utility to view and select from a list of available unattached virtual disks.

    The BIG-IP system creates a virtual disk with a default name (the guest name plus the string `.img`, such as `guestA.img`).

12. For the **VLAN List** setting, select both an internal and an external VLAN name from the **Available** list, and use the Move button to move the VLAN names to the **Selected** list.

    The VLANs in the **Available** list are part of the vCMP host configuration.

    After you create the guest, the guest can use the selected VLANs to process application traffic.

13. From the **Requested State** list, select **Provisioned**.
    After the guest is created, the vCMP host allocates all necessary resources to the guest, such as cores and virtual disk.

14. If you want to enable Appliance mode for the guest, select the **Appliance Mode** check box.

    Warning: Before enabling this feature on an isolated guest, you must perform some prerequisite tasks, such as creating a self IP address on the guest. Failure to perform these prerequisite tasks will make the guest unreachable by all host and guest administrators. Therefore, you must create the isolated guest with Appliance mode disabled, perform the prerequisite tasks, and then modify the guest to enable this setting. For more information, see the relevant appendix of this guide.

    When you enable **Appliance Mode** for a guest, the system enhances security by denying access to the `root` account and the `Bash` shell for all administrators.

15. Click **Finish**.
    The system installs the selected ISO image onto the guest's virtual disk and displays a status bar to show the progress of the resource allocation.

You now have a new vCMP guest on the system that a guest administrator can access to provision and configure BIG-IP modules.

Setting a vCMP guest to the Deployed state

Setting a guest to the Deployed state enables a guest administrator to then provision and configure the BIG-IP® modules within the guest.

Warning: For any isolated guest with Appliance mode enabled, you must first perform some additional tasks before deploying the guest. For more information, see the relevant appendix of this guide.

---

1. Ensure that you are logged in to the vCMP host.
2. On the Main tab, click **vCMP** > **Guest List**.
   This displays a list of vCMP guests on the system.
3. In the Name column, click the name of the vCMP guest that you want to deploy.
4. From the **Requested State** list, select **Deployed**.
5. Click **Update**.

After moving a vCMP® guest to the Deployed state, a guest administrator can provision and configure the BIG-IP modules within the guest so that the guest can begin processing application traffic.

vCMP guest administrator tasks

The primary duties of a vCMP® guest administrator are to provision BIG-IP® modules within the guest and configure any self IP addresses that the guest needs for processing application traffic. The guest administrator must also configure all BIG-IP modules, such as creating virtual servers and load balancing pools within BIG-IP Local Traffic Manager™ (LTM).

Optionally, a guest administrator who wants a redundant system configuration can create a device group with the peer guests as members.

Provisioning BIG-IP modules within a guest

Before a guest administrator can access a guest to provision licensed BIG-IP® modules, the vCMP® guest must be in the Deployed state.

To run BIG-IP modules within a guest, the guest administrator must first provision them. For example, a guest administrator for guestA who wants to run LTM® and GTM™ must log into guestA and provision the LTM and GTM modules.

Note: For guests that are isolated from the management network, you must access them using a self IP address instead of a management IP address.

---

1. Open a browser, and in the URL field, specify the management IP address that the host administrator assigned to the guest.
2. At the login prompt, type the default user name admin, and password admin, and click **Log in**.
   The Setup utility screen opens.
3. Click **Next**.

This displays the Resource Provisioning screen.

4. For each licensed BIG-IP module in the list, select the check box and select **Minimal**, **Nominal**, or **Dedicated**.

5. Click **Next**.
This displays the Certificate Properties screen.

6. Click **Next**.
This displays some general properties of the guest.

7. Click **Next**.
This displays the screen for specifying the guest's cluster member IP addresses.

8. Click **Next**.

9. Click **Finished**.

Creating a self IP address for application traffic

A vCMP® guest administrator creates a self IP address within a guest, assigning a VLAN to the address in the process. The self IP address serves as a hop for application traffic destined for a virtual server configured within the guest. On a standalone system, the self IP address that a guest administrator creates is a static (non-floating) IP address. Note that the administrator does not need to create VLANs within the guest; instead, the VLANs available for assigning to a self IP address are VLANs that a host administrator previously created on the vCMP host.

1. On the Main tab of the BIG-IP Configuration utility, click **Network** > **Self IPs**.
The Self IPs screen opens.

2. Click **Create**.
The New Self IP screen opens.

3. In the **Name** field, type a unique name for the self IP.

4. In the **IP Address** field, type an IPv4 or IPv6 address.
This IP address should represent the address space of the VLAN that you specify with the **VLAN/Tunnel** setting.

5. In the **Netmask** field, type the network mask for the specified IP address.

6. From the **VLAN/Tunnel** list, select the VLAN to associate with this self IP address.

   • On the internal network, select the VLAN that is associated with an internal interface or trunk.
   • On the external network, select the VLAN that is associated with an external interface or trunk.

7. From the **Port Lockdown** list, select **Allow Default**.

8. Click **Finished**.
The screen refreshes, and displays the new self IP address.

After creating a self IP address, the BIG-IP system can send and receive traffic destined for a virtual server that allows traffic through the specified VLAN.

Next steps

After all guests are in the Deployed state, each individual guest administrator can configure the appropriate BIG-IP modules for processing application traffic. For example, a guest administrator can use BIG-IP® Local Traffic Manager™ (LTM®) to create a standard virtual server and a load-balancing pool. Optionally, if guest redundancy is required, a guest administrator can set up device service clustering (DSC®).

Another important task for a guest administrator is to create other guest administrator accounts as needed.

Important: *If the guest has an isolated (rather than bridged) management network, you must grant access to the Traffic Management Shell (`tmsh`) to all guest administrator accounts. Otherwise, guest administrators have no means of logging in to the guest, due to the lack of access to the management network.*

Configuration results

After you and all guest administrators have completed the initial configuration tasks, you should have a system provisioned for vCMP, with one or more guests ready to process application traffic.

When logged in to the vCMP host, you can see the VLANs and trunks configured on the system, as well as all of the guests that you created, along with their virtual disks. You can also see the number of cores that the host allocated to each guest.

When logged in to a guest, the guest administrator can see one or more BIG-IP® modules provisioned and configured within the guest to process application traffic. If the guest administrator configured device service clustering (DSC®), the guest is a member of a device group.

vCMP Core Upgrade for Deployed Guests

About increasing CPU cores for a deployed guest

Overview

If you have a guest that's actively processing application traffic but needs additional CPU cores, you can increase the number of cores that are currently allocated to the guest. To do this, you'll need to temporarily disable the guest while you add cores, and then re-deploy the guest.

About CPU core availability

When you increase the **Cores per Guest** value on a guest, the new value must be within the total number of unused CPU cores on the system; the system never allocates more cores to a guest than the number of cores still available on the system.

For example, suppose you have three guests on a 12-core system, with this core allocation:

Guest Name	Current Cores per Guest
Guest A	6
Guest B	2
Guest C	2

With a total allocation of 10 out of 12 cores allocated on the system, this means that only two cores remain unused and available to add to a guest. Therefore, if you try to add four additional cores to **Guest C** (from **2** to **6**), the system displays a message that you are exceeding the number of cores available on the system. In this case, the maximum number of cores that you can allocate to **Guest C** is **4**.

A potential workaround for this would be to first decrease the number of cores allocated to **Guest A**, to free up additional cores for **Guest C**.

You can determine the number of cores still available on the system by logging in to the BIG-IP Configuration utility and displaying the vCMP Guest List screen.

Increasing CPU cores for a deployed guest

Before doing this task, confirm that your BIG-IP® user account has an Administrator role assigned to it.

If you have a running vCMP guest, and you decide that you need to allocate more CPU cores to the guest than what was originally allocated, you can increase the number of cores for the guest.

To do this, you'll first need to set the guest state from **Deployed** to **Configured**. Once you've increased the number of cores for the guest, you can set the guest back to the **Deployed** state.

Note that when switching between guest states, the process can take several minutes for the guest to shut down cleanly and restart again.

Important: *When you set a guest to the **Configured** state, the guest is removed from service until you set the guest back to the **Deployed** state.*

---

1. Log in to the BIG-IP Configuration utility on the vCMP host, using the BIG-IP system's primary management IP address.
2. On the Main tab, click **vCMP** > **Guest List**.
3. View the graphic to determine the number of unused cores available on the system.
   This tells you how many cores are available to add to the guest.
4. In the Name column, find the name of the guest that needs additional CPU cores and confirm that the Requested State is **Deployed**.
5. To the left of the guest name, select the check box.
6. Click the **Disable** button.
   The guest state changes from **Deployed** to **Configured**.

   Important: *It might take a few minutes for the guest state to change to **Configured**.*

   ---

7. Once the screen shows that the guest is now in the **Configured** state, click the guest name.
   The guest properties screen opens.
8. From the **Cores per Guest** list, select a higher number of CPU cores.

   Note: *Do not select a value that exceeds the total number of unused cores available for use on the slot.*

   For example, if you currently have two cores allocated to the guest and you want to add two cores, then from the **Cores per Guest** list, select **4**.
9. From the **Requested State** list, select **Deployed**.
10. Click **Update**.
    The Guest List screen opens again and you will see that the guest state is changing.

    Important: *It might take a few minutes for the guest state to change back to the **Deployed** state.*

After you complete this task, the guest has additional CPU cores and is capable of processing application traffic.

Managing vCMP Virtual Disks

Overview: Managing vCMP virtual disks

A *virtual disk* is the portion of disk space that the system has allocated to a guest. Each virtual disk is implemented as an image file with an .img extension, such as guest_A.img.

The vCMP® system automatically creates a virtual disk when you create a vCMP guest. However, after you have created and deployed all guests, you can manage the virtual disks on the system as a way to optimize performance. You must be logged into the vCMP host to manage virtual disks on the system.

Using the BIG-IP® Configuration utility or the Traffic Management Shell (tmsh), you can delete virtual disks on the system as a way to optimize disk space.

About virtual disk allocation

On the vCMP system, the host assigns a fixed amount of disk space (100 gigabytes) for each guest. This amount of available disk space per guest remains fixed regardless of the number of cores that you configure for that guest. For example, allocating two cores to guest_A provides the same amount of available disk space for the guest as allocating four cores to the guest.

Note that you cannot explicitly create virtual disks; instead, the BIG-IP® system creates virtual disks when the guest changes to a Provisioned or Deployed state. You can create a guest that remains in the Configured state, but in this case, the guest has no virtual disk allocated to it.

About virtual disk images

A virtual disk is in the form of an image that resides in the /shared/vmdisks directory. The default file name that the BIG-IP® system initially assigns to a virtual disk is the guest name plus a .img extension (for example, guestA.img). You identify and manage virtual disks on the system using these file names.

About virtual disk detachment and re-attachment

When a vCMP® guest has no virtual disk and moves from the Configured state to the Provisioned state, the system creates a virtual disk and attaches the disk to the guest. This attachment ensures that only that guest can use the virtual disk. A guest can have only one virtual disk attached to it at any one time.

A virtual disk can become unattached from a guest when you perform one of these actions:

- Delete a guest.
- Change the **Virtual Disk** property of the guest to **None**. Note that to perform this action, you must first change the guest state to Configured.

With either of these actions, the system retains the virtual disks on the system for future use.

You can attach an existing, unattached virtual disk to a new guest that you create. Attaching an existing virtual disk to a newly-created guest saves the BIG-IP® system from having to create a new virtual disk for the guest.

Detaching virtual disks from a vCMP guest

Before you can detach a virtual disk from a guest, you must be logged into the vCMP host. Also, you must change the **Requested State** property on the guest to **Configured**.

You can detach a virtual disk from the guest, but retain the virtual disk on the BIG-IP® system so that you can attach it to another guest later.

Important: *Unattached virtual disks consume disk space on the system. To prevent unattached virtual disks from depleting available disk space, routinely monitor the number of unattached virtual disks that exist on the system.*

---

1. On the Main tab, click **vCMP** > **Guest List**.
   This displays a list of vCMP guests on the system.
2. In the Name column, locate the relevant guest name, and to the left of the name, select the check box.
3. Click the **Configured** button.
4. In the Name column, click the guest name.
5. From the **Virtual Disk** list, select the default value, **None**.
6. Click **Update**.

The vCMP guest no longer has any virtual disk attached to it.

Viewing virtual disks not attached to a vCMP guest

Before you can view unattached virtual disks, you must be logged into the vCMP host.

You can view virtual disks that are not attached to a vCMP® guest so that you can monitor virtual disks that might be unused but still consuming disk space.

1. On the Main tab, click **vCMP** > **Virtual Disk List**.
2. Locate the Virtual Disk List area of the screen.
3. To the right of the list of virtual disk names, note any disks that do not have any guest names associated with them. These disks are unattached.

Attaching a detached virtual disk to a vCMP guest

Before you begin this task, ensure that:

- You are logged into the vCMP® host.
- The guest to which you are attaching the virtual disk is in the Configured state.
- The virtual disk is not currently be attached to another guest.

It is possible for a virtual disk to become detached from a vCMP guest. A disk that is no longer attached to a guest is known as an *unattached virtual disk*.

You can attach an unattached virtual disk to another guest either when you create the guest or when you modify the **Virtual Disk** property of a guest.

1. On the Main tab, click **vCMP** > **Guest List**.

This displays a list of vCMP guests on the system.

2. In the Name column, click the name of the vCMP guest that you want to view.

3. From the **Properties** list, select **Advanced**.

4. From the **Virtual Disk** list, select a file name.
   The guest uses the newly-selected virtual disk when being deployed.

5. Click **Update**.

Deleting a virtual disk from the BIG-IP system

Before deleting a virtual disk, ensure that you are logged into the vCMP host.

Using the BIG-IP® Configuration utility, you can delete a virtual disk from the system.

Important: This is the only way to delete a virtual disk from the system. If you delete the associated guest instead, the system retains the virtual disk for re-use by another guest later.

1. On the Main tab, click **vCMP** > **Virtual Disk List**.

2. Locate the Virtual Disk List area of the screen.

3. In the Name column, locate the name of the virtual disk that you want to delete.

4. To the left of the virtual disk name, select the check box.

5. Click **Delete**.
   The system prompts you to confirm the delete action.

6. Click **Delete**.

Viewing vCMP Statistics

Overview: Viewing vCMP statistics

After creating vCMP® guests to process application traffic, you can display vCMP guest statistics to better manage performance.

Viewing virtual disk statistics

Before viewing virtual disk statistics, you must be logged in to the vCMP host.

Using the BIG-IP® Configuration utility, you can view information about the virtual disks that are currently allocated to vCMP® guests:

- The virtual disk names
- The size in gigabytes of each virtual disk
- The name of the guest to which each virtual disk is currently allocated

1. On the Main tab, click **vCMP** > **Virtual Disk List**.
2. Locate the Virtual Disk List area of the screen.

The following table shows sample statistics for three separate virtual disks.

Virtual Disk Name	Operating System	Status	Disk use
GuestA.img	TMOS	Ready	64.4G
GuestB.img	Unknown	Unknown	64.4G
GuestC.img	TMOS	Ready	64.4G

Viewing vCMP guest statistics with the BIG-IP Configuration utility

Before viewing guest statistics, you must be logged in to the vCMP host.

Using the BIG-IP® Configuration utility, you can list the names of, and information about, the vCMP® guests that are currently on the system.

1. Log out of the guest.
2. On an external system, open a browser window and access the vCMP host, using the vCMP host's management IP address.
3. Using your user credentials, log in to the BIG-IP Configuration utility.
4. On the Main tab, click **vCMP**.

The system displays a list of vCMP guest names, as well as this information:

- The state configured for the guest
- The guest's management IP address and netmask

Viewing disk usage statistics

Before viewing disk usage statistics, you must be logged in to the vCMP host.

Using the BIG-IP® Configuration utility, you can view information about the vCMP® disk usage:

- Disk name
- The number of virtual disks
- The total vCMP application volume size, in gigabytes
- The available vCMP application volume size, in gigabytes

1. On the Main tab, click **vCMP** > **Virtual Disk List**.
2. Locate the Disk Usage area of the screen.

Additional Tasks for Isolated Guests in Appliance Mode

Additional tasks for isolated guests in Appliance mode

To ensure that guest administrators can access an isolated guest and manage the BIG-IP® software within the guest, you must create the isolated guest with Appliance mode disabled, perform some additional tasks, and then modify the guest to enable Appliance mode. These additional tasks are:

- Creating a self IP address for guest administrators to use to access the guest, and granting `tmsh` access to the guest's `admin` user account.
- Enabling Appliance mode on the guest.

After performing these tasks, administrators for an isolated guest are restricted to using either the BIG-IP® Configuration utility or `tmsh` to manage BIG-IP modules within the guest (when port lockdown settings on the self IP address allow such traffic).

Preparing an isolated guest for Appliance mode

You use this task to prepare an isolated guest to operate in Appliance mode. Specifically, you use this task to:

- Grant access to the Traffic Management Shell (`tmsh`) for the `admin` user account within a vCMP® guest. Because the `admin` user for an isolated guest in Appliance mode is restricted to using `tmsh`, you must first grant the `admin` account permission to use `tmsh`. By default, the `admin` account for a guest has no access to `tmsh`.
- Create a self IP address for guest administrators to use to access the guest. This is necessary because an isolated guest is not connected to the management network and therefore has no management IP address assigned to it.

You perform this task by accessing the guest from the vCMP® host.

1. From the vCMP host, access the Bash shell by typing `vconsole` *guest_name*.
   For example, you can type `vconsole guest_A`
   The system prompts you to enter a user name and password.
2. Log in to the guest using the `root` account and the password `default`.
   A system prompt is displayed.
3. Type the command `tmsh modify auth user admin shell tmsh`.
   This command grants `tmsh` access to the `admin` user account.
4. Type the command `tmsh create net self address` *ip_address/netmask* `vlan` *vlan_name* `allow-service default`.
   This creates the specified IP address on the guest and makes required adjustments to the port lockdown settings.
5. At the prompt, exit the guest by typing `exit`.
6. Exit the vConsole utility by typing the key sequence `ctrl-]`.
   This displays the prompt `telnet>`.
7. Type `q`.

Enabling Appliance mode on an isolated guest

You use this task to enable Appliance mode on an existing guest that is isolated from the management network.

Note: You can perform this task while the guest is in the Deployed or Provisioned state; there is no need to set the guest state to Configured prior to performing this task.

1. Use a browser to log in to the vCMP® host, using the primary cluster management IP address.

   Note: If you provisioned the system for vCMP®, this step logs you in to the vCMP host.

2. On the Main tab, click **vCMP** > **Guest List**.
   This displays a list of vCMP guests on the system.

3. In the Name column, click the name of the guest that you want to modify.
   This displays the configured properties of the guest.

4. For the **Appliance Mode** setting, select the check box.
   When you enable **Appliance Mode** for an isolated guest, the system enhances security by denying access to the `root` account and the `Bash` shell for all guest administrators.

5. Click **Update**.

The guest is now running in Appliance mode. All guest administrators are restricted to using the BIG-IP® Configuration utility and `tmsh` to manage the guest.

Index

Index