F5 BigIP (LTM) Quick Start Guide for Hitachi Content Platform (HCP)

Jeff Stewart (Hitachi Vantara)

Contents

[Introduction 2](#_Toc1378505)

[Overview 3](#_Toc1378506)

[Design Criteria: 5](#_Toc1378507)

[Conceptual Diagrams: 6](#_Toc1378508)

[Solution Design Requirements 9](#_Toc1378509)

[DNS Reservations: 9](#_Toc1378510)

[SSL Certificate Management: 10](#_Toc1378511)

[HCP Design Requirements: 11](#_Toc1378512)

[HCP Provisioning Standards: 13](#_Toc1378513)

[Configure Services 14](#_Toc1378514)

[Step 1: Create F5 traffic Monitors for each HCP Service 15](#_Toc1378515)

[HCP-DATA-Monitor: 15](#_Toc1378516)

[HCP-MQE-Monitor: 16](#_Toc1378517)

[HCP-NSB-Monitor: 17](#_Toc1378518)

[Step 2: Create F5 traffic Pools for each HCP Service 18](#_Toc1378519)

[HCP-DATA-Pool: 18](#_Toc1378520)

[HCP-MQE-Pool: 19](#_Toc1378521)

[HCP-NSB-Pool: 20](#_Toc1378522)

[Step 3: Create F5 Virtual Servers for each HCP Service – on each HCP Tenant 21](#_Toc1378523)

[HCP-DATA-VirtualServer: 21](#_Toc1378524)

[HCP-MQE-VirtualServer: 23](#_Toc1378525)

[HCP-NSB-VirtualServer: 24](#_Toc1378526)

[Appendix 25](#_Toc1378527)

# Introduction

This document is intended as a reference architecture guide to configure and deploy an F5 BigIP (LTM) load balancer with Hitachi Content Platform (HCP).

This quick start guide makes the following assumptions:

* Both the F5 and HCP systems have been setup and are in their initial state of functional deployment according to customer standards.
* Network (VLAN) connectivity between the F5 and HCP systems is available based on customer standards and best practices (F5 systems to be placed as close as possible to the HCP (~hop-count wise - same access switches, subnet, low latency connectivity, etc).
* Best practice recommendation from Hitachi will always call for the F5 BigIP solution to be *wholly dedicated* to the HCP it is ‘front-ending’… But in any deployment mode, care should be taken to size for adequate ‘front-end’ resources on the selected F5 BigIP platform based on the ‘back-end’ workload delivery capabilities of the HCP.

# Overview

This reference architecture outlines the configuration steps required to implement an F5 BigIP (LTM) load balancer in front of HCP based on a combination of Hitachi recommended best practices and customer specific application requirements.

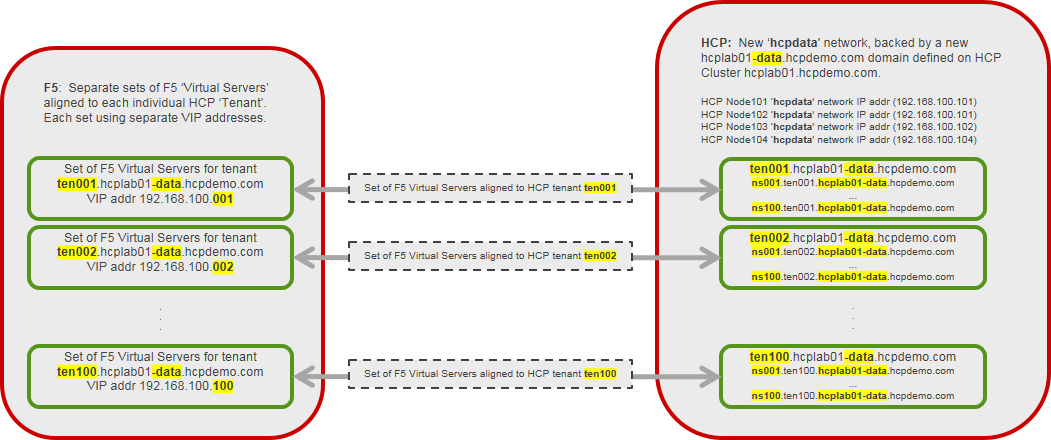
**Notes on design:** In order to meet various customer requirements, the solution described here is based on a one-to-one ‘**logical**’ association between an F5 ‘Virtual Server’ and an HCP ‘Tenant’ for data access and related operations. In actuality, a ‘set’ of F5 Virtual Servers will be created for each HCP Tenant – one Virtual Server for each of the major services that will be front-ended through the load balancer for a single HCP Tenant. However, the different sets of Virtual Servers will all leverage a common set of F5 traffic ‘Monitors’ and traffic ’Pools’ that will be aligned with the entire HCP ‘Cluster’.

Only ‘data access’ related operations and services on an HCP Tenant will be front-ended through the load balancer. All client applications will be routed to the load balancer to use these services via new ‘virtualized’ tenant and namespace names (tenant and namespace URL’s) in DNS. These new DNS names will include a “-data” string inserted into the real HCP domain name and will resolve to the Virtual IP (VIP) address assigned to each ‘set’ of F5 Virtual Servers to be associated with a given HCP Tenant.

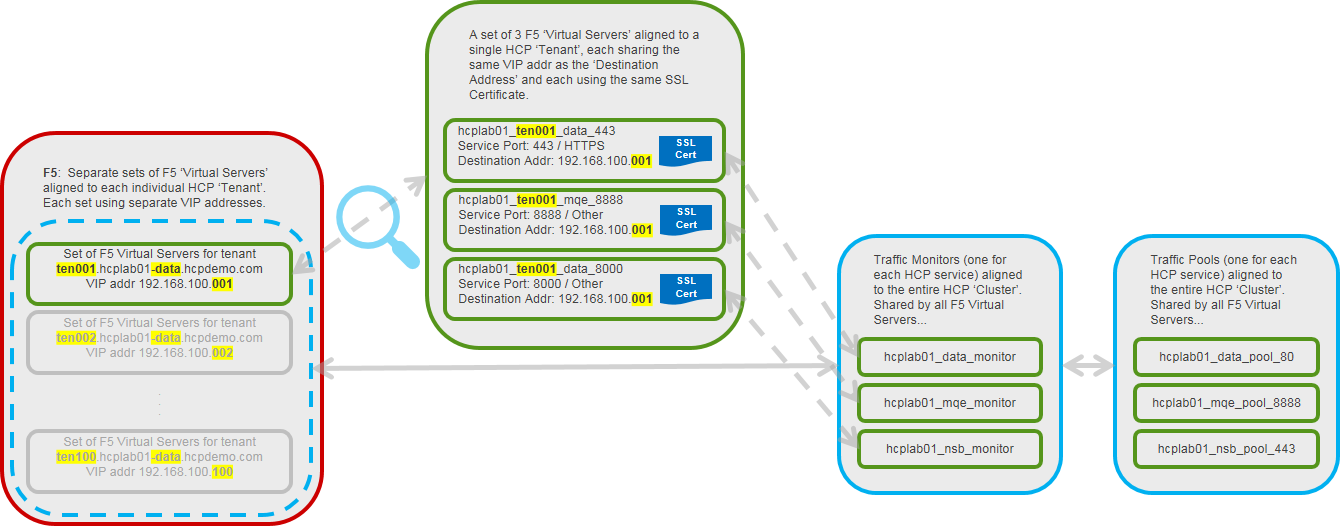
All ‘systems management’ related operations and services on HCP should continue to be routed directly to the HCP via the existing DNS implementation (i.e. bypassing the load balancer).

This design allows for separation of ‘data access’ and ‘systems management’ traffic for fault tolerance purposes (i.e. systems management is not dependent on connectivity to the load balancer).

Example illustration of multiple ‘sets’ of F5 Virtual Servers and their association to multiple HCP Tenants:



As described above, each ‘set’ of F5 Virtual Servers that align to a given HCP Tenant represented above is in fact comprised of 3 individual Virtual Servers (one for each of the HCP services on the associated Tenant that will be front-ended by the load balancer).



In this design, those services will be for HTTPS Data Access, the Search Console (for *tenant level* MQE only), and the Namespace Browser (NSB).

**NOTE:** Each of the 3 F5 Virtual Servers associated with HCP Tenant ‘ten001’ shown above are configured to provide access to the different services they support over different IP ports. The different Virtual Servers and the services and ports they support are:

* hcplab01\_ten001\_**data\_443**: HTTPS Data Access (**DATA** access, over port **443**)
* hcplab01\_ten001\_**mqe\_8888**: Metadata Query Engine (Search) Console (**MQE**, over port **8888**)
* hcplab01\_ten001\_**nsb\_8000**: Namespace Browser (**NSB**, over port **8000**)

**NOTE:** As an SSL offload based design - Data Access takes place over HTTPS from client to load balancer (port 443), but then communications between load balancer and HCP takes place over HTTP (port 80). This is otherwise referred to as ‘terminating’ SSL at the load balancer.

**NOTE:** For Namespace Browser (NSB) access, because SSL connectivity must be maintained between BOTH client and load balancer as well as between load balancer and HCP, in order to distinguish NSB access from HTTPS data requests – port redirection is used to offer NSB access to clients via HTTPS over port 8000. The load balancer will establish it’s own HTTPS (standard port 443) session to HCP (this is required by HCP to allow the browser session).

*Also important to note with regards to the NSB – this document does NOT take into account session persistence and/or the use of cookies required to keep NSB sessions active.*

Each Virtual Server for Tenant ‘ten001’ shares the same VIP address (each F5 Virtual Servers ‘Destination Address’ is set to 192.168.100.**001**). In this way, clients and applications have a single ‘logical’ VIP to connect to for any of the services on the given HCP Tenant (ten001) being front-ended by the load balancer.

Each of the Virtual Servers associated with all Tenants on the HCP will leverage a common set of F5 traffic Monitors and traffic Pools that will align with the entire HCP Cluster. Sharing a common set of traffic Monitors and Pools across all Virtual Servers in this way serves to reduce the overall amount of monitor traffic and pool member selection / decision making activity between the load balancer and the HCP.

## Design Criteria:

* **Need for SSL Certificate management**: due to limitations in HCP’s SSL certificate management features compared to the granularity of management required, the F5 solution is ideal for SSL certificate and traffic offload.

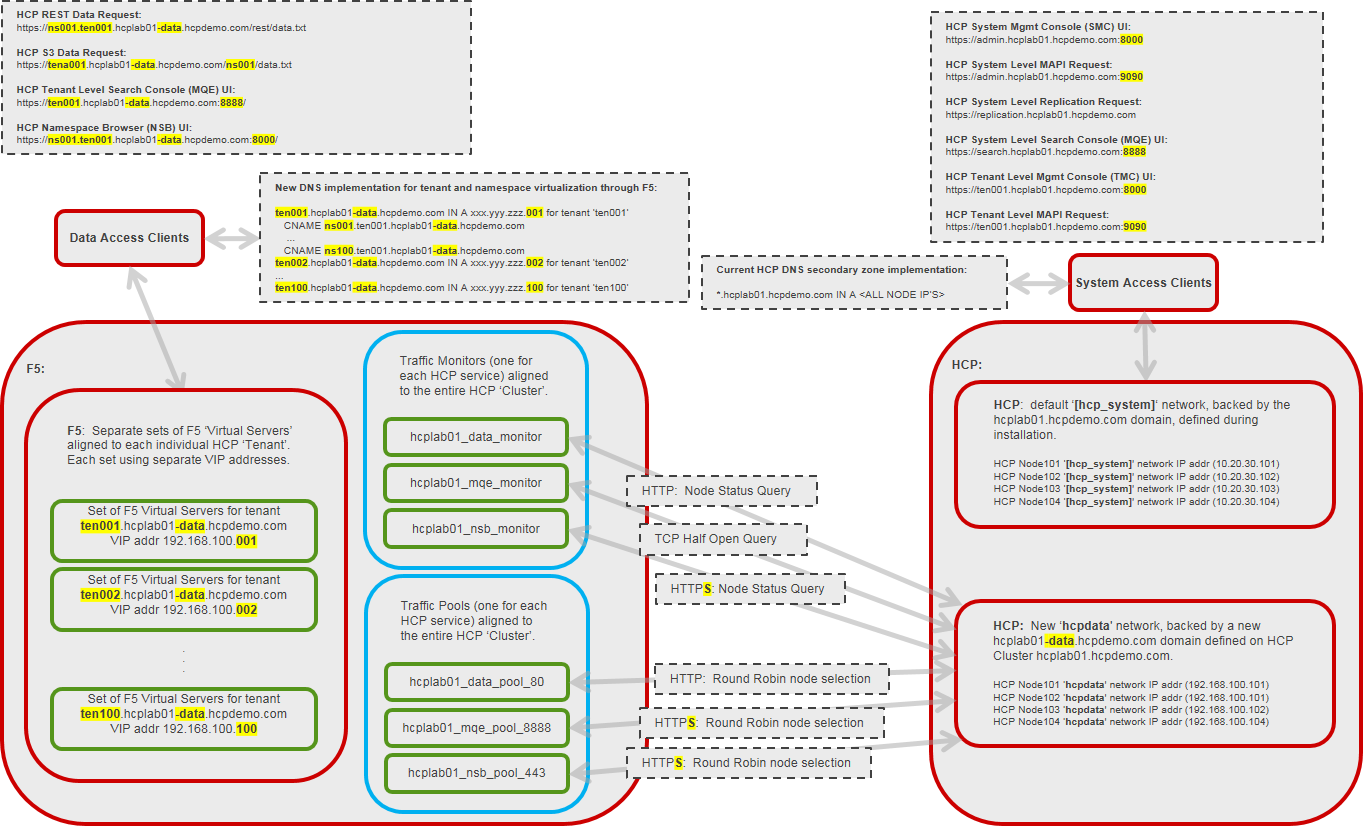
**NOTE about SSL:** the design described here has SSL/HTTPS being ‘**terminated**’ at the load balancer for DATA access. This is to isolate the SSL Certificate management challenges to the load balancer layer at a virtual HCP Tenant level (i.e. one SSL certificate per virtualized HCP Tenant to be loaded at the load balancer on all Virtual Servers supporting a given HCP Tenant).

* **A single solution that would accommodate the use of both Native REST and S3 protocols with HCP:** With the customers strategic direction for the use of object storage being focused at standardizing on the S3 protocol, but with current applications handling interop with HCP via the native REST protocol, a load balancer solution is required that will accommodate BOTH protocols’ use-cases, as well as being compatible with Amazon AWS v4 signature authentication (which can be difficult to accommodate on a multi-tenant object storage platform).
* **Allowance for rate limiting policies**: This architecture will reserve for ***future consideration***, the use of standard rate limiting policies on F5 BigIP, which may be applied at the Virtual Server level. With the design described here being based on a one-to-one ‘***logical***’ association between an F5 ‘Virtual Server’ and an HCP ‘Tenant’, some care would need to be taken to not oversubscribe too many F5 Virtual Servers to a single multi-tenant HCP.
* **Application specific telemetry requirements**: This architecture will reserve for ***future consideration***, the use of advanced F5 BigIP features to preserve the following client/application specific attributes and data points that are of interest and concern to some applications:
* **Original Client/Application Source IP address**: F5 standard/default behavior in a Layer 7 SNAD deployment is to preserve the originating client IP address and store it in the standard “X-Forwarded-For” (XFF) header. This is expected to be a common requirement for all application use-cases, particularly for troubleshooting purposes on shared HCP’s where it will be important to distinguish between different originating client IP addresses.

**Notes on sizing:** Any candidate HCP architecture being considered will require assessment and sizing to ensure the front-end F5 BigIP resources are kept adequate to support the ‘back-end’ HCP workload potential… Best Practice recommendation is to always use ***dedicated*** load balancers with HCP.

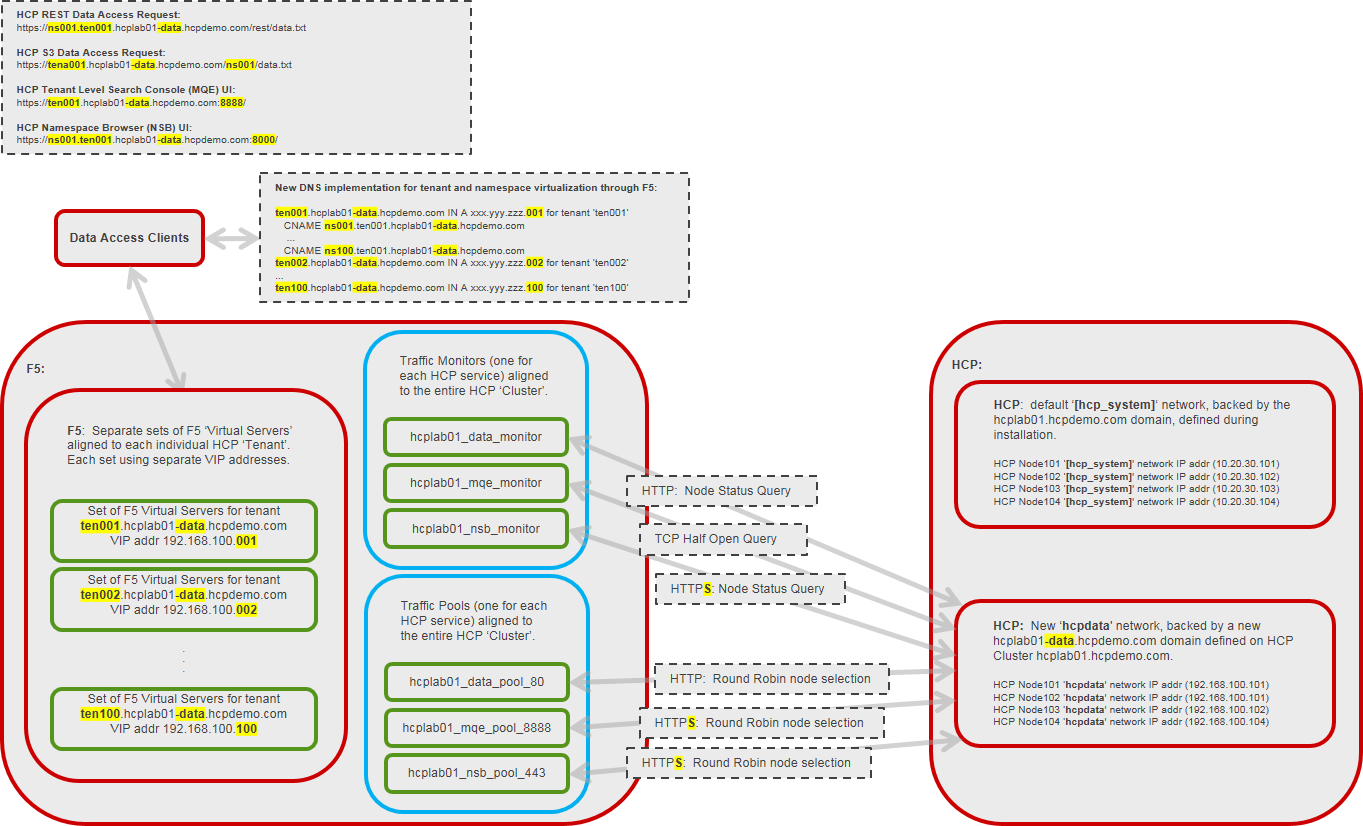
## Conceptual Diagrams:

This diagram illustrates the separation of ‘data access’ requests (which will be routed through the load balancer via DNS) from ‘systems management’ requests (which will be routed directly to the HCP via the existing DNS implementation - bypassing the load balancer). Note that SSL/HTTPS is also being terminated at the load balancer for DATA access:

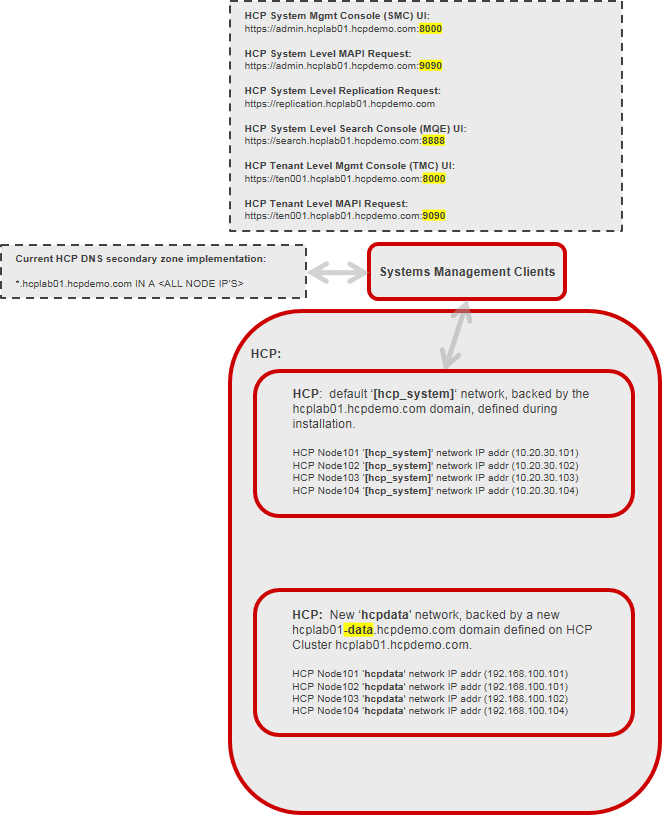


The diagram also illustrates how each HCP Tenant being virtualized by F5 leverages a common set of F5 traffic Monitors and traffic Pools.

This diagram focuses on how Data requests, MQE and NSB sessions will be routed through the new ‘**hcpdata**’ network via the new DNS implementation described in this design document:



This diagram focuses on how Systems management (SMC), System level Replication and Search (MQE), Tenant management (TMC), and both System and Tenant level MAPI requests will continue to be routed through the default ‘**[hcp\_system]**’ network and existing DNS implementation:



Load Balancer Network Map:

# Solution Design Requirements

## DNS Reservations:

The design described here requires several additional DNS reservations to be made beyond that which is normally put in place for HCP. The recommendations made here take into consideration the restriction imposed by the customer against the use of wildcards in DNS:

**NOTE:** The restriction against the use of wildcards has a knock-on effect of requiring that we statically pre-allocate resources names (tenant and namespace names) so that we can issue DNS requests up front with all of the names that *will be* allowed for each virtualized tenant. This is mainly to reduce management burden.

* When an HCP ‘Cluster’ is deployed, a supporting F5 Load Balancer should be deployed as well.
* When an HCP ‘Tenant’ is provisioned for a Line of Business or Application, a DNS reservation to virtualize the Tenant and all its Namespace URL’s (limited to ns001 – ns100 in this design) should be made that resolves to the VIP address that will be assigned to the set of F5 Virtual Servers that will be associated with the HCP Tenant:
  + Actual DNS Name for the back-end HCP: **hcplab01.hcpdemo.com**
  + DNS Name (Host A Record) to resolve the virtualized HCP Tenant URL to the VIP address to be assigned to the F5 Virtual Server(s) for the associated HCP Tenant:
    - Host Record FQDN: **ten001.hcplab01-data.hcpdemo.com**
  + DNS CNAMES under this Host A Record to resolve the virtualized HCP tenant’s Namespace URL’s to the VIP address to be assigned to the F5 Virtual Server(s) for the associated Tenant:
    - CNAME: **ns001.ten001.hcplab01-data.hcpdemo.com**
    - CNAME: **ns002.ten001.hcplab01-data.hcpdemo.com**
    - CNAME: **ns003.ten001.hcplab01-data.hcpdemo.com**
    - ...
    - CNAME: **ns100.ten001.hcplab01-data.hcpdemo.com**

**NOTE:** In order to limit the number of entries in both DNS and the SSL certificates required, this design will impose an artificial maximum limit of 100 tenants per HCP and 100 Namespaces per Tenant.

## SSL Certificate Management:

The design described here allows for the separation of management of SSL Certificates (a primary driver for introducing the load balancer), while remaining within other limitations that have influenced the design:

* Restrictions imposed by the customer against the use of vendor platform ‘self-signed’ certificates, in favor of internally generated certificates (to be expired and renewed according to customer internal policy).
* Restrictions imposed by the customer against the use of wildcards in SSL ‘Subject Alternative Names’ (SSL certificate SANs) in customer generated SSL certificates.
  + This has a knock-on effect of requiring that the customer statically pre-allocate resources names (tenant and namespace URL’s) so that the customer can issue certificates with all of the resource names for each virtualized tenant up front, to reduce management burden.

Working within these limitations allows for the use of an SSL Certificate with explicit SANs for all of the resources for a given HCP Tenant – to be loaded onto the F5 Virtual Servers front-ending the given HCP Tenant. As an example:

[ dn ]

CN = ten001.hcplab01**-data**.hcpdemo.com

[ req\_ext ]

subjectAltName = @alt\_names

[ alt\_names ]

DNS.1 = ns001.ten001.hcplab01**-data**.hcpdemo.com

DNS.2 = ns002.ten001.hcplab01**-data**.hcpdemo.com

...

DNS.100 = ns100.ten001.hcplab01**-data**.hcpdemo.com

This will allow for an SSL Certificate management strategy that will limit any potential impact to applications as a result of activities such as frequent SSL Certificate updates to only those users and applications sharing a single HCP Tenant – which as a matter of customer provisioning standards is often aligned with a single Line of Business or Application.

**Notes on provisioning workflow and SSL Certificate management responsibility:**

* The provisioning team will need to submit two infrastructure requests when provisioning new HCP Tenants:
  + DNS and IP request for the VIP of the F5 Virtual Servers to front-end the new HCP Tenant(s) and Namespace(s).
  + An F5 request for the new Virtual Servers to be provisioned with the names from the DNS request. The F5 admins will need to provision the appropriate F5 resources, and request for an appropriate SSL Certificate to be generated and then loaded on the new Virtual Servers for each Tenant.

## HCP Design Requirements:

The design described here calls for HCP configuration changes that differ from basic deployment standards:

* An additional ‘**Domain**’ must be defined on the HCP to be associated with the ‘virtualized’ HCP Tenant names used when accessing HCP Tenants through the load balancer. For example:
  + Actual DNS Name for the HCP: **hcplab01.hcpdemo.com**
  + Default ‘Domain’ defined on the HCP: **hcplab01.hcpdemo.com**
  + New ‘Domain’ defined on the HCP: **hcplab01-data.hcpdemo.com**
* An additional ‘**Network**’ must be defined on the HCP that will be used as the data access (‘**hcpdata**’) network for all HCP Tenants to be front-ended by the load balancer. This must be a different VLAN on the same physical network, but different subnet as the current front-end ‘**[hcp\_system]**’ network.
  + **NOTE:** this requires enabling the ‘**Configuration / Advanced Settings / Enable virtual network management**’ option in HCP.
  + An additional IP address on the VLAN for each HCP node to support the new ‘**hcpdata**’ network will be required.
  + The new ‘**hcpdata**’ network will be leveraged *ONLY* for Data access to the HCP Tenants and Namespaces, including the MQE Console (Search) and Namespace Browser services.
  + The default ‘**[hcp\_system]**’ network will continue to be leveraged for System management (SMC), System level Replication and Search, Tenant management (TMC), and both System and Tenant level MAPI services.
* A customer generated SSL Certificate to allow for secure management of the System and Tenants over the HCP ‘**[hcp\_system]**’ Network. The recommendation here would be for the creation of a single SSL certificate for all Tenants to be allowed for provisioning. As an example:

**NOTE:** This is for the certificate to be loaded directly onto the HCP under the ‘<**hcp-cluster-name>.<domain\_name>**’ Domain associated with the default/existing ‘**[hcp\_system]**’ Network:

[ dn ]

CN = hcplab01.hcpdemo.com

[ req\_ext ]

subjectAltName = @alt\_names

[ alt\_names ]

DNS.1 = admin.hcplab01.hcpdemo.com

DNS.2 = replication.hcplab01.hcpdemo.com

DNS.3 = search.hcplab01.hcpdemo.com

DNS.4 = ten001.hcplab01.hcpdemo.com

DNS.5 = ten002.hcplab01.hcpdemo.com

...

DNS.103 = ten100.hcplab01.hcpdemo.com

* An additional customer generated SSL Certificate to allow for secure access to Tenant level resources via the HCP ‘**hcpdata**’ Network. The recommendation here would be for the creation of a single SSL certificate for all Tenants to be allowed for provisioning. As an example (modeled after the previous certificate for simplicity):

**NOTE:** This is for the certificate to be loaded directly onto the HCP under the new ‘**<hcp-cluster-name>-data.<domain\_name>**’ Domain backing the new ‘**hcpdata**’ Network. Note the inclusion of the “**-data**” string in the otherwise normal HCP cluster name representing the ‘virtualized’ HCP resources:

[ dn ]

CN = hcplab01**-data**.hcpdemo.com

[ req\_ext ]

subjectAltName = @alt\_names

[ alt\_names ]

DNS.1 = admin.hcplab01**-data**.hcpdemo.com

DNS.2 = replication.hcplab01**-data**.hcpdemo.com

DNS.3 = search.hcplab01**-data**.hcpdemo.com

DNS.4 = ten001.hcplab01**-data**.hcpdemo.com

DNS.5 = ten002.hcplab01**-data**.hcpdemo.com

...

DNS.103 = ten100.hcplab01**-data**.hcpdemo.com

## HCP Provisioning Standards:

The design described here has the following implications on HCP provisioning guidelines:

* Use of the Namespace Browser directly to the HCP will be restricted.
* HCP Tenants should have their **Management Network** set to ‘**[hcp\_system]**‘ and **Data Network** set to ‘**hcpdata**’.

# Configure Services

For each HCP service on a single HCP Cluster that will be front-ended by F5 (Management, MAPI, and System level Replication and Search services will bypass the load balancer and go direct to the HCP), we will create an F5 traffic Monitor and traffic Pool:

* HCP-DATA (to support HTTPS based access to data through the load balancer)
* HCP-MQE (to support MQE ‘Search’ Console access through the load balancer)
* HCP-NSB (to support Namespace Browser access through the load balancer)

For each HCP Tenant on an HCP, a ‘set’ of F5 Virtual servers will also be created, one Virtual Server to be associated with each of the HCP services mentioned above.

However, as previously described - where the F5 traffic Monitors and Pools align to an entire HCP Cluster, a ‘set’ of F5 Virtual Servers in this design align with an individual HCP Tenant.

## Step 1: Create F5 traffic Monitors for each HCP Service

Create an F5 traffic Monitor for each of the HCP Services to be front-ended by F5.

**NOTE:** F5 traffic Monitors will be aligned with an entire HCP ‘Cluster’. This means they only get created ONCE when a new HCP Cluster is brought online.

### HCP-DATA-Monitor:

For the HCP-DATA-Monitor, create a new traffic Monitor called **<hcp-cluster-name>\_data\_monitor**. For example:

* Actual DNS Name for the HCP: **hcplab01.hcpdemo.com**
* Name for the HCP-DATA-Monitor on F5: **hcplab01\_data\_monitor**

For the **Type** select **HTTP**.

**NOTE:** Because this design ‘terminates’ SSL and HTTPS at the load balancer, HCP itself will be configured for HTTP access (at a per namespace level).

Leave **Interval** set to **5** seconds (this can be made more aggressive to suit application requirements with engineering approval).

Leave **Timeout** set to **16** seconds (again, this can be made more aggressive to suit application requirements with engineering approval).

The **Send String** should be set to “**GET /node\_status HTTP/1.0\r\n\r\n**”. This is the HTTP(s) string that will be sent to the HCP nodes to determine overall node and data access availability.

The **Receive String** should be set to **204**. An HTTP 204 response means the node is in good health. The rest of the values can be left at their defaults.

Example(s):

Monitor Properties:

To-do: insert screenshot of F5 ‘**Local Traffic >> Monitors >> hcplab01\_data\_monitor**‘ Properties page...

### HCP-MQE-Monitor:

For the HCP- MQE-Monitor, create a new traffic Monitor called **<hcp-cluster-name>\_mqe\_monitor**. For example:

* Actual DNS Name for the HCP: **hcplab01.hcpdemo.com**
* Name for the HCP-MQE-Monitor on F5: **hcplab01\_mqe\_monitor**

For the **Type** select **TCP Half Open**.

**NOTE:** The MQE service on HCP has no explicit service specific status query to indicate service availability. For this reason, we will simply rely on a basic TCP monitor to assess MQE status and availability.

Leave **Interval** set to **5** seconds (this can be made more aggressive to suit application requirements with engineering approval).

Leave **Timeout** set to **16** seconds (again, this can be made more aggressive to suit application requirements with engineering approval).

Example(s):

Monitor Properties:

To-do: insert screenshot of F5 ‘**Local Traffic >> Monitors >> hcplab01\_mqe\_monitor**‘ Properties page...

### HCP-NSB-Monitor:

For the HCP- NSB-Monitor, create a new traffic Monitor called **<hcp-cluster-name>\_nsb\_monitor**. For example:

* Actual DNS Name for the HCP: **hcplab01.hcpdemo.com**
* Name for the HCP-NSB-Monitor on F5: **hcplab01\_nsb\_monitor**

For the **Type** select **HTTPS**.

**NOTE:** This is intentionally different than the HCP-Data-Monitor configuration, and is intended to preserve SSL communication from the load balancer to the HCP when using HCP’s Namespace Browser through the load balancer.

Leave **Interval** set to **30** seconds (this can be made more aggressive to suit application requirements with engineering approval).

Leave **Timeout** set to **91** seconds (again, this can be made more aggressive to suit application requirements with engineering approval).

The **Send String** should be set to “**GET /node\_status HTTP/1.0\r\n\r\n**”. This is the HTTP(s) string that will be sent to the HCP nodes to determine overall node and data access availability.

The **Receive String** should be set to **204**. An HTTP 204 response means the node is in good health. The rest of the values can left as default.

Example(s):

Monitor Properties:

To-do: insert screenshot of F5 ‘**Local Traffic >> Monitors >> hcplab01\_nsb\_monitor**‘ Properties page...

## Step 2: Create F5 traffic Pools for each HCP Service

Create an F5 traffic Pool for each of the HCP Services to be front-ended by F5.

**NOTE:** F5 traffic Pools will be aligned with an entire HCP ‘Cluster’. This means they only get created ONCE when a new HCP Cluster is brought online.

### HCP-DATA-Pool:

For the HCP-DATA-Pool, create a new traffic Pool called **<hcp-cluster-name>\_data\_pool\_80**. For example:

* Actual DNS Name for the HCP: **hcplab01.hcpdemo.com**
* Name for the HCP-DATA-Pool on F5: **hcplab01\_data\_pool\_80**

Add the **<hcp-cluster-name>\_data\_monitor** traffic Monitor created in Step 1 as the Active **Health Monitor** for this pool.

Set **Load Balancing Method** to **Round Robin**.

Add each HCP nodes front-end ‘**hcpdata**’ Network IP address as members of the pool, with **Service Port** set to **80** (similarly to setting the HCP-DATA-Monitor as an HTTP monitor, this is because we are ‘terminating’ SSL and HTTPS at the load balancer).

Example(s):

Pool Properties:

To-do: insert screenshot of F5 ‘**Local Traffic >> Pools : Pools List >> hcplab01\_data\_pool\_80**‘ Properties page...

Pool Members:

To-do: insert screenshot of F5 ‘**Local Traffic >> Pools : Pools List >> hcplab01\_data\_pool\_80**‘ Members page...

Individual Pool Member Properties:

To-do: insert screenshot of F5 ‘**Local Traffic >> Pools : Pools List >> hcplab01\_data\_pool\_80**‘ Members page, after clicking on one of the Current Members of the pool...

### HCP-MQE-Pool:

For the HCP-MQE-Pool, create a new traffic Pool called **<hcp-cluster-name>\_mqe\_pool\_8888**. For example:

* Actual DNS Name for the HCP: **hcplab01.hcpdemo.com**
* Name for the HCP-MQE-Pool on F5: **hcplab01\_mqe\_pool\_8888**

Set the **<hcp-cluster-name>\_mqe\_monitor** traffic Monitor created in Step 1 as the Active **Health Monitor** for this pool.

Set **Load Balancing Method** to **Round Robin**.

Add each HCP node’s front-end ‘**hcpdata**’ Network IP address as members of the pool, with **Service Port** set to **8888** (this is the native port on which the MQE Console runs on HCP).

Example(s):

Pool Properties:

To-do: insert screenshot of F5 ‘**Local Traffic >> Pools : Pools List >> hcplab01\_mqe\_pool\_8888**‘ Properties page...

Pool Members:

To-do: insert screenshot of F5 ‘**Local Traffic >> Pools : Pools List >> hcplab01\_mqe\_pool\_8888**‘ Members page...

Individual Pool Member Properties:

To-do: insert screenshot of F5 ‘**Local Traffic >> Pools : Pools List >> hcplab01\_ mqe \_pool\_8888**‘ Members page, after clicking on one of the Current Members of the pool...

### HCP-NSB-Pool:

For the HCP-NSB-Pool, create a new traffic Pool called **<hcp-cluster-name>\_nsb\_pool\_443**. For example:

* Actual DNS Name for the HCP: **hcplab01.hcpdemo.com**
* Name for the HCP-NSB-Pool on F5: **hcplab01\_nsb\_pool\_443**

Set the **<hcp-cluster-name>\_nsb\_monitor** traffic Monitor created in Step 1 as the Active **Health Monitor** for this pool.

Set **Load Balancing Method** to **Round Robin**.

Add each HCP node’s front-end ‘**hcpdata**’ Network IP address as members of the pool, with **Service Port** set to **443** (this is to preserve HTTPS communication between the load balancer and HCP).

Example(s):

Pool Properties:

To-do: insert screenshot of F5 ‘**Local Traffic >> Pools : Pools List >> hcplab01\_nsb\_pool\_443**‘ Properties page...

Pool Members:

To-do: insert screenshot of F5 ‘**Local Traffic >> Pools : Pools List >> hcplab01\_nsb\_pool\_443**‘ Members page...

Individual Pool Member Properties:

To-do: insert screenshot of F5 ‘**Local Traffic >> Pools : Pools List >> hcplab01\_ nsb \_pool\_443**‘ Members page, after clicking on one of the Current Members of the pool...

## Step 3: Create F5 Virtual Servers for each HCP Service – on each HCP Tenant

Create an F5 Virtual Server for each of the HCP Services to be front-ended by F5, for a single HCP Tenant.

Virtual Servers on F5 BigIP provide the point of connectivity for applications to access HCP Tenants through the load balancer.

**NOTE:** As referenced throughout this document, the different HCP services being front-ended through the F5 load balancer operate over distinct IP ports, which current provisioning standards require to be associated with distinct virtual servers on the load balancer. So while we will in fact create THREE virtual servers for each HCP Tenant, we will refer to them as a single ‘logical’ Virtual Server since all the virtual servers for a given HCP Tenant will share the same ‘Destination Address’.

### HCP-DATA-VirtualServer:

For the HCP-DATA-VirtualServer, create a new Virtual Server called **<hcp-cluster-name>\_<hcp-tenant-name>\_data\_443**. For example:

* Actual DNS Name for the HCP: **hcplab01.hcpdemo.com**
* HCP Tenant: **ten001**
* Name for the HCP-DATA-Virtual Server 1 on F5: **hcplab01\_ten001\_data\_443**

For the **Type** select **Standard**.

Enter the client/application facing VIP Address allocated on the VLAN for the F5 Virtual Server associated with the HCP Tenant being virtualized as the **Destination Address** and set the **Service Port** to **443** / **HTTPS**.

**NOTE:** The remaining virtual servers will be configured to use the SAME VIP / **Destination Address** as the HCP-DATA-VirtualServer being created here. This will effectively present a single ‘logical’ Virtual Server for clients and applications to connect to through the load balancer for all services on the given HCP Tenant.

Set **Protocol** to **TCP**.

Set **Protocol Profile (Client)** to **tcp-wan-optimized**.

Set **Protocol Profile (Server)** to **tcp-lan-optimized**.

Set **HTTP Profile** to **http-xff**.

Set **SSL Profile (Client)** to **<hcp-cluster-name>\_clientssl**.

Set **VLAN and Tunnel Traffic** to **All VLANs and Tunnels**.

Set **Source Address Translation** to **Auto Map** if your F5 is deployed in a single arm configuration so that the return traffic passes back through the F5.

Set the **Default Pool** to the <**hcp-cluster-name>\_data\_pool\_80**’ traffic Pool created in Step 2.

Finally, set **OneConnect Profile** to **oneconnect**.

Example(s):

Virtual Server Properties:

To-do: insert screenshot of F5 ‘**Local Traffic >> Virtual Servers : Virtual Server List >> hcplab01\_ten001\_data\_443**‘ Properties page...

Virtual Server Resources:

To-do: insert screenshot of F5 ‘**Local Traffic >> Virtual Servers : Virtual Server List >> hcplab01\_ten001\_data\_443**‘ Resources page...

### HCP-MQE-VirtualServer:

For the HCP-MQE-VirtualServer, create a new Virtual Server called **<hcp-cluster-name>\_<hcp-tenant-name>\_mqe\_8888**. For example:

* Actual DNS Name for the HCP: **hcplab01.hcpdemo.com**
* HCP Tenant: **ten001**
* Name for the HCP-MQE-VirtualServer on F5: **hcplab01\_ten001\_mqe\_8888**

For the **Type** select **Standard**.

Enter the client/application facing VIP Address allocated on the VLAN for the F5 Virtual Server associated with the HCP Tenant being virtualized as the **Destination Address** and set the **Service Port** to **8888** / **Other**.

Set **Protocol** to **TCP**.

Set **Protocol Profile (Client)** to **tcp-wan-optimized**.

Set **Protocol Profile (Server)** to **tcp-lan-optimized**.

Set **HTTP Profile** to **http-xff**.

Set **SSL Profile (Client)** to **<hcp-cluster-name>\_clientssl**.

Set **SSL Profile (Server)** to **<hcp-cluster-name>\_serverssl**.

Set **VLAN and Tunnel Traffic** to **All VLANs and Tunnels**.

Set **Source Address Translation** to **Auto Map** if your F5 is deployed in a single arm configuration so that the return traffic passes back through the F5.

Set the **Default Pool** to the <**hcp-cluster-name>\_mqe\_pool\_8888**’ traffic Pool created in Step 2.

Finally, set the **Default Persistence Profile** to **cookie**.

Example(s):

Virtual Server Properties:

To-do: insert screenshot of F5 ‘**Local Traffic >> Virtual Servers : Virtual Server List >> hcplab01\_ten001\_mqe\_8888**‘ Properties page...

Virtual Server Resources:

To-do: insert screenshot of F5 ‘**Local Traffic >> Virtual Servers : Virtual Server List >> hcplab01\_ten001\_mqe\_8888**‘ Resources page...

### HCP-NSB-VirtualServer:

For the HCP-NSB-VirtualServer, create a new traffic Virtual Server called **<hcp-cluster-name>\_tenant<#>\_nsb\_8000**. For example:

* Actual DNS Name for the HCP: **hcplab01.hcpdemo.com**
* HCP Tenant: **ten001**
* Name for the HCP-NSB-VirtualServer on F5: **hcplab01\_ten001\_nsb\_8000**

For the **Type** select **Standard**.

Enter the client/application facing VIP Address allocated on the VLAN for the F5 Virtual Server associated with the HCP Tenant being virtualized as the **Destination Address** and set the **Service Port** to **8000** / **Other**.

Set **Protocol** to **TCP**.

Set **Protocol Profile (Client)** to **tcp-wan-optimized**.

Set **Protocol Profile (Server)** to **tcp-lan-optimized**.

Set **HTTP Profile** to **http-xff**.

Set **SSL Profile (Client)** to **<hcp-cluster-name>\_clientssl**.

Set **SSL Profile (Server)** to **<hcp-cluster-name>\_serverssl**.

Set **VLAN and Tunnel Traffic** to **All VLANs and Tunnels**.

Set **Source Address Translation** to **Auto Map** if your F5 is deployed in a single arm configuration so that the return traffic passes back through the F5.

Finally, set the **Default Pool** to the <**hcp-cluster-name>\_nsb\_pool\_443**’ traffic Pool created in Step 2.

Example(s):

Virtual Server Properties:

To-do: insert screenshot of F5 ‘**Local Traffic >> Virtual Servers : Virtual Server List >> hcplab01\_ten001\_nsb\_8000**‘ Properties page...

Virtual Server Resources:

To-do: insert screenshot of F5 ‘**Local Traffic >> Virtual Servers : Virtual Server List >> hcplab01\_ten001\_nsb\_8000**‘ Resources page...

# Appendix