cloudera®

Impala HA with F5 BIG-IP

Important Notice

Cloudera, Inc.
1001 Page Mill Road, Building 2
Palo Alto, CA 94304-1008
info@cloudera.com
US: 1-888-789-1488
Intl: 1-650-843-0595
www.cloudera.com

Release Information

Date: 20160414

Table of Contents

Overview

This guide walks you through configuring an F5 BIG-IP to manage client connection traffic to Apache Impala (incubating) traffic using Local Traffic Manager (LTM), providing high availability and protecting against Impala daemon failures. Step-by-step instructions are provided to configure LTM to work with ODBC, JDBC, impala-shell, and Hue, both with and without TLS/SSL enabled.

Known issues and limitations are discussed.

This guide does not address performance tuning of Impala or LTM.

We find that users often use the terms F5, BIG-IP, and LTM interchangeably. This usage is not accurate and can lead to confusion. Think of the BIG-IP as the physical hardware appliance sitting in the datacenter, running a specially crafted operating system to manage network traffic. Local Traffic Manager (LTM) is one of the modules running on that OS, providing traffic optimization, load balancing, and offloading.

The vast majority of this guide refers to LTM. Some of the configuration described -- such as loading certificates -- affects the BIG-IP as a whole, not just the LTM. A few operations pertain to Cloudera Manager and the cluster hosts themselves.

Instructions

Prerequisites

- An F5 BIG-IP configured with functional network interfaces, trunks, VLANs, and routes required to pass traffic
- A self-IP added to BIG-IP networking for the Impala services, or available ports on an existing one
- An existing cluster running Impala
- A fully qualified domain name (FQDN) that resolves to the self-IP

Basic understanding of general load balancing concepts is recommended, especially before making changes to production systems. Links to the F5 Glossary and BIG-IP LTM manual are provided for F5-specific terms.

Basics

Impala uses two "external" TCP ports, each of which must be configured in a similar manner:

- 21000 - Frontend (impala-shell, Beeswax, v1.2 of Cloudera ODBC)
- 21050 - Frontend (Hue, JDBC, v2.0+ of Cloudera ODBC)

The Impala Daemon HTTP Server runs on TCP port 25000. You do not need to configure this port for load balancing; users should go to the host directly.

The StateStore and Catalog server ports are also listed as external, but only run on a single host. Do not configure these services for load balancing.

There are four steps when configuring an Impala service on an LTM:

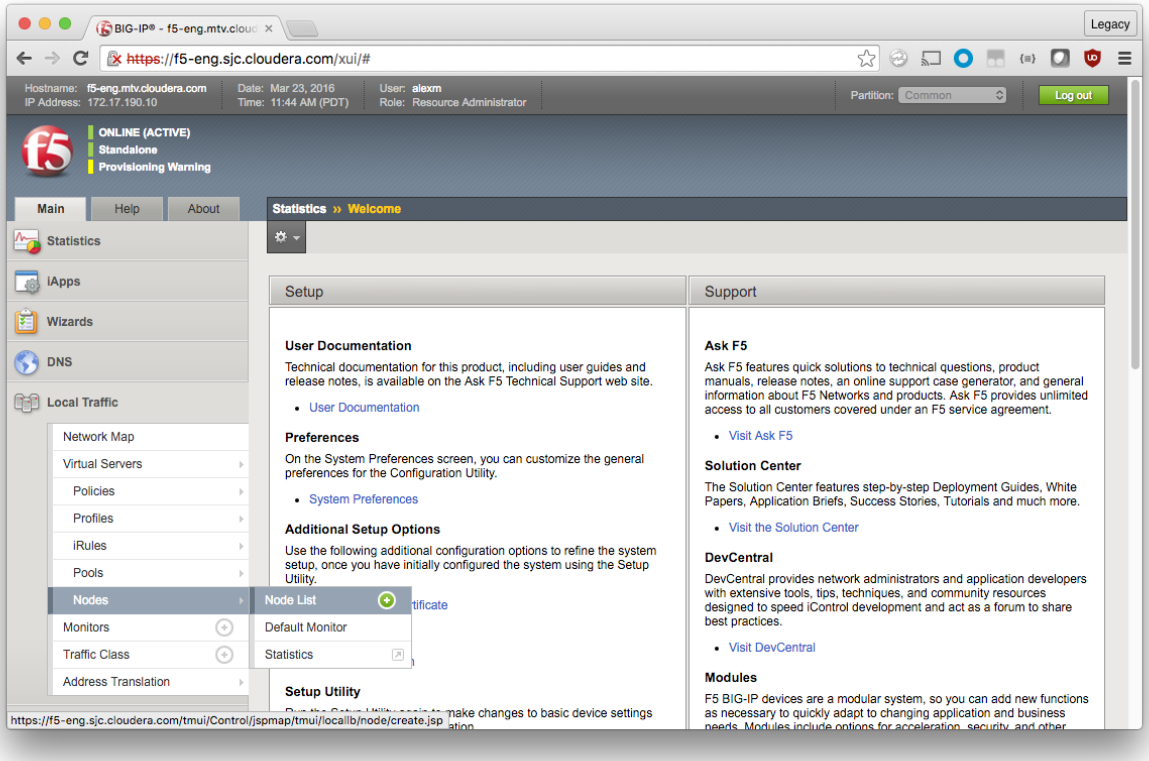
1. Create the nodes.
2. Create the pools.
3. Create the virtual servers.
4. Configure Impala to be used through a proxy for high availability.

Depending on your need for TLS/SSL, additional configuration may be required. Once you have the basics, see Supplemental Configuration.

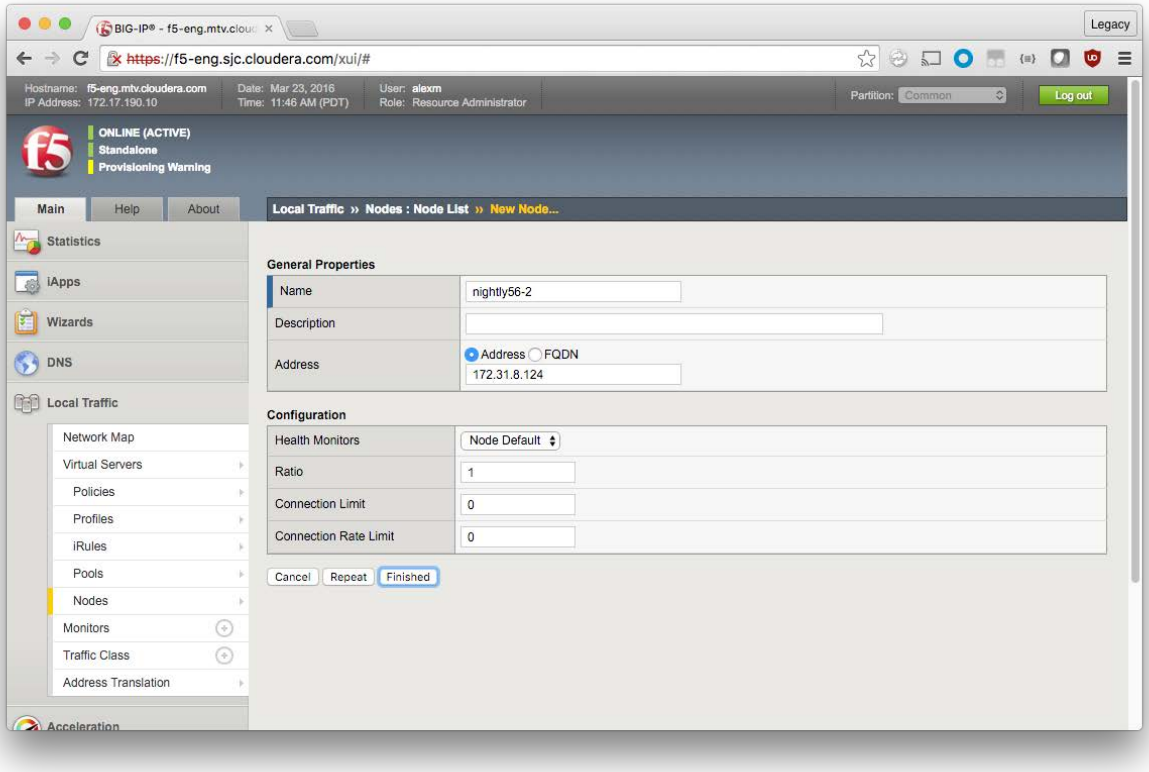
Create the Nodes

For each cluster host that is targeted by the load balancer, you create a node record on the load balancer.

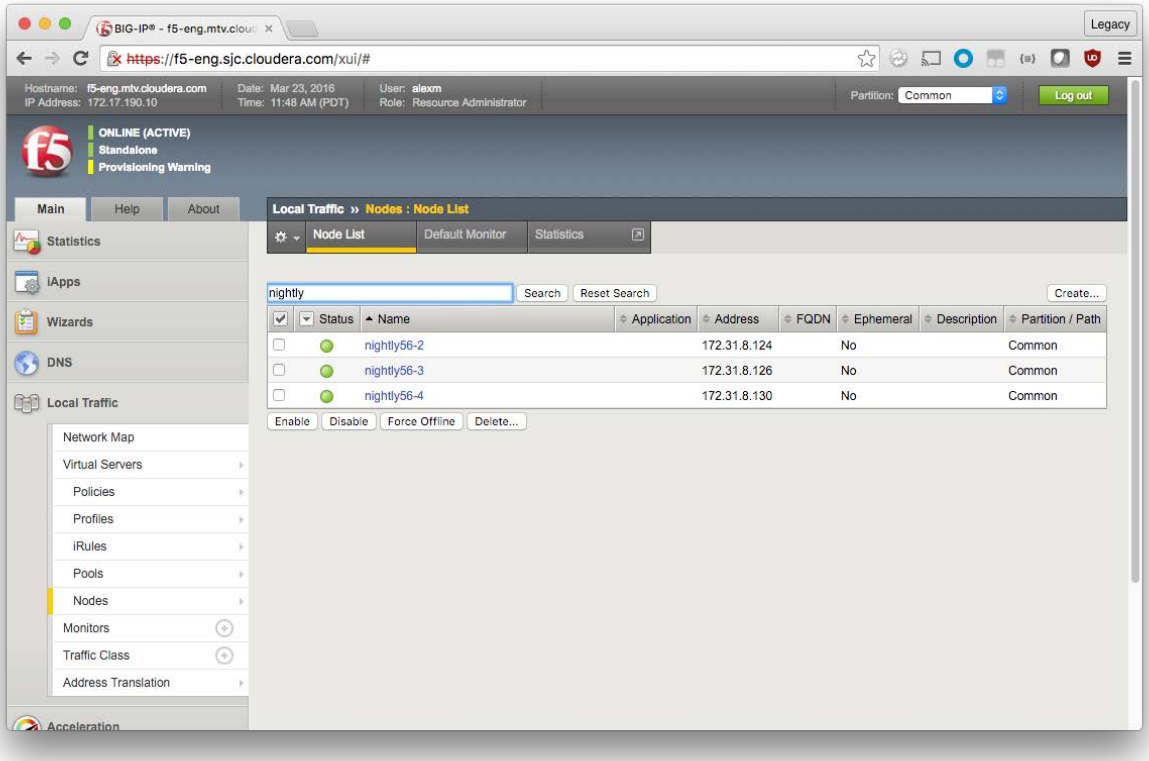
Local Traffic > Nodes > Node List > Create (green plus)



Assign each node a descriptive label and IP address. Although you can use a fully qualified domain name (FQDN), Cloudera recommends using the IP address of the node instead of relying on name resolution.



Repeat until all of your Impala nodes are defined.

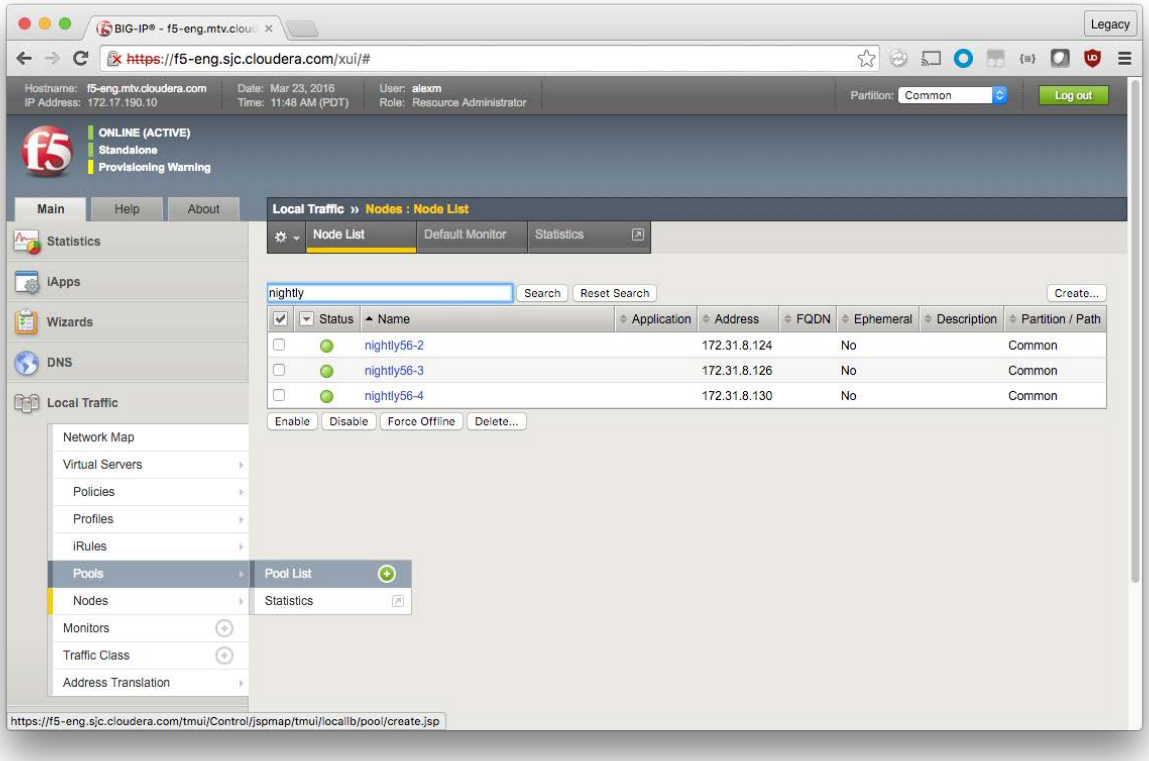


Create the Pools

A pool represents a collection of hosts running a service to be balanced according to a particular mechanism. Impala has two frontend services, each of which need a distinct pool.

Port 21000

Local Traffic > Pools > Pool List > Create (green plus)



Use the following settings for the pool.

Configuration (Basic)

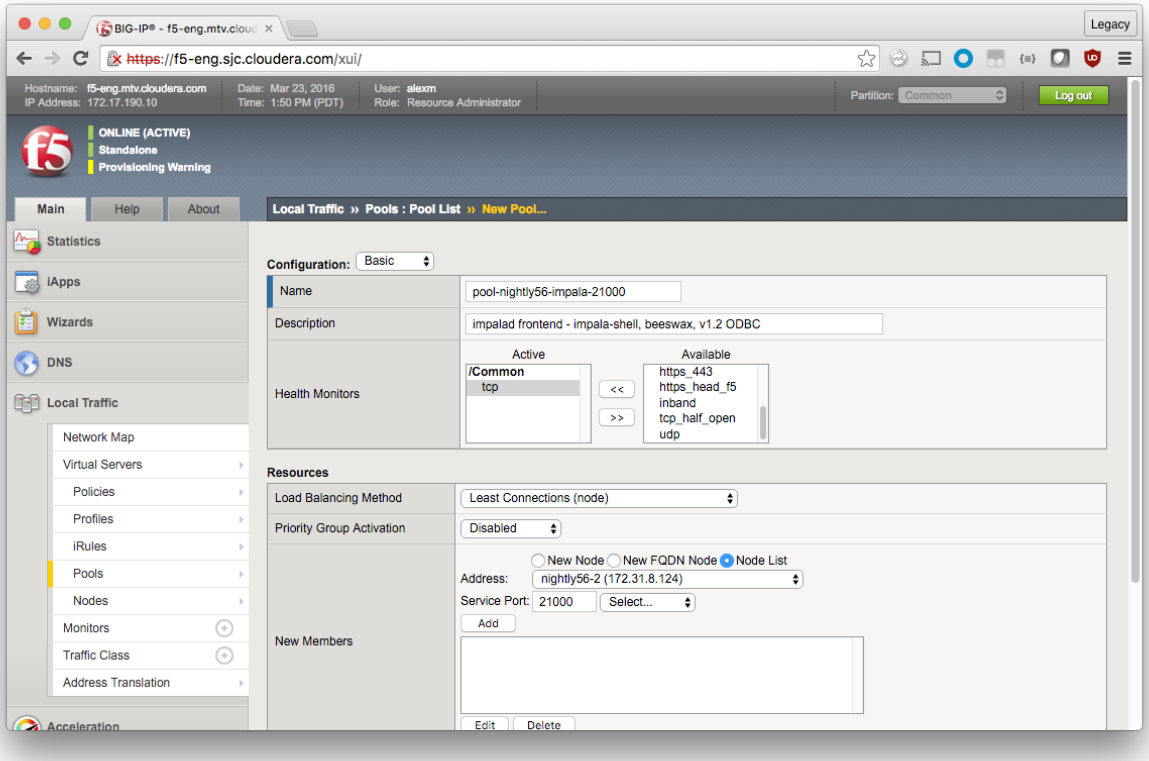
Name: pool-nightly56-impala-21000

Description: impalad frontend - impala-shell, beeswax, v1.2 ODBC

Health Monitors: tcp

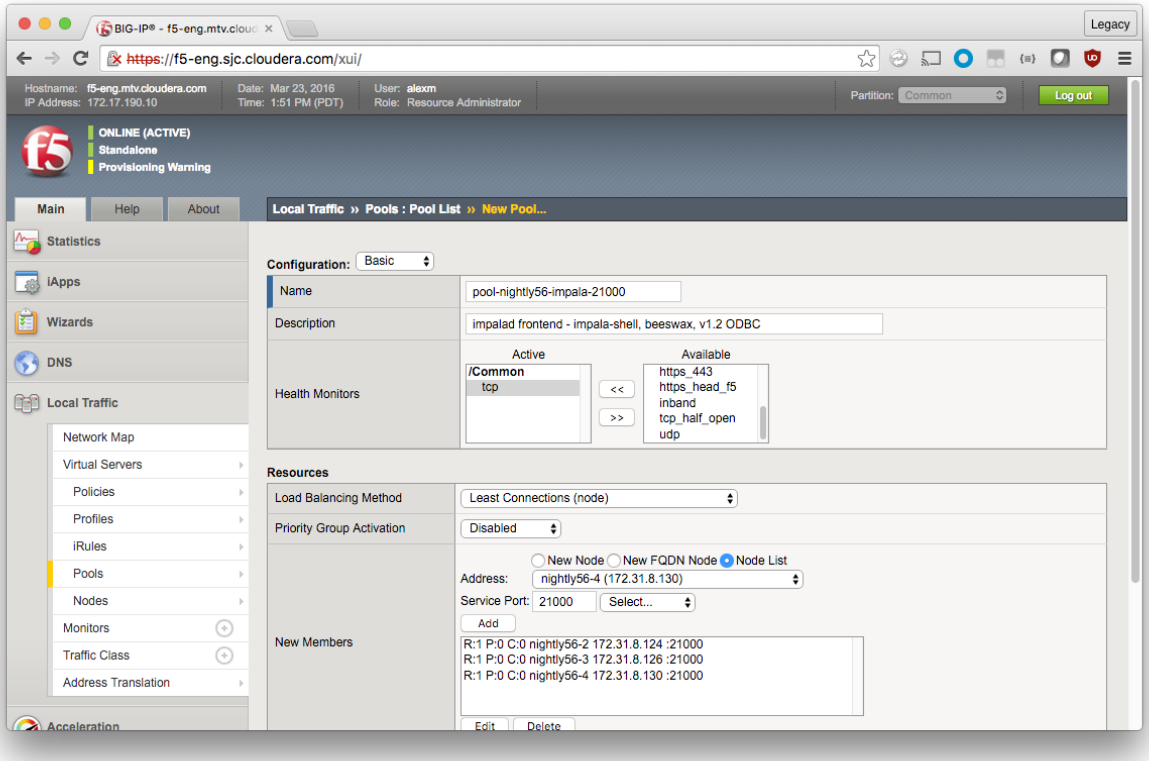
Resources

Load Balancing Method: Least Connections (node)



In the **New Members** section:

1. Click the **Node List** radio button.
2. Select the first node, assign the **Service Port** as 21000.
3. Click **Add**.
4. Repeat for each Impala node.



Click **Finished**.

Follow the same process for the next pool.

Port 21050

Local Traffic > Pools > Pool List > Create (green plus)

Use the following settings for the pool.

Configuration (Basic)

Name: pool-nightly56-impala-21050
Description: impalad frontend - JDBC, v2.0+ of Cloudera ODBC
Health Monitors: tcp

Resources

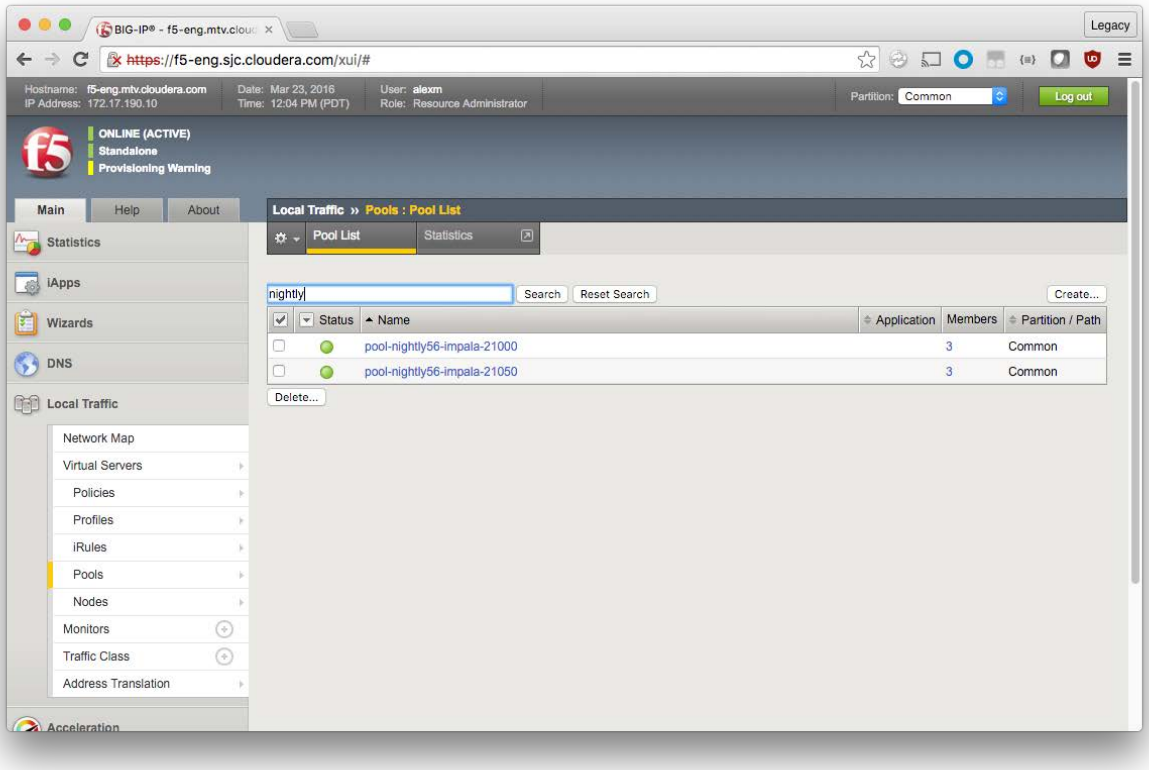
Load Balancing Method: Least Connections (node)

In the **New Members** section

1. Click the **Node List** radio button.
2. Select the first node, assign the **Service Port** as 21050.
3. Click **Add**.
4. Repeat for each Impala node.

Click **Finished**.

You now see the two pools, each with three members.

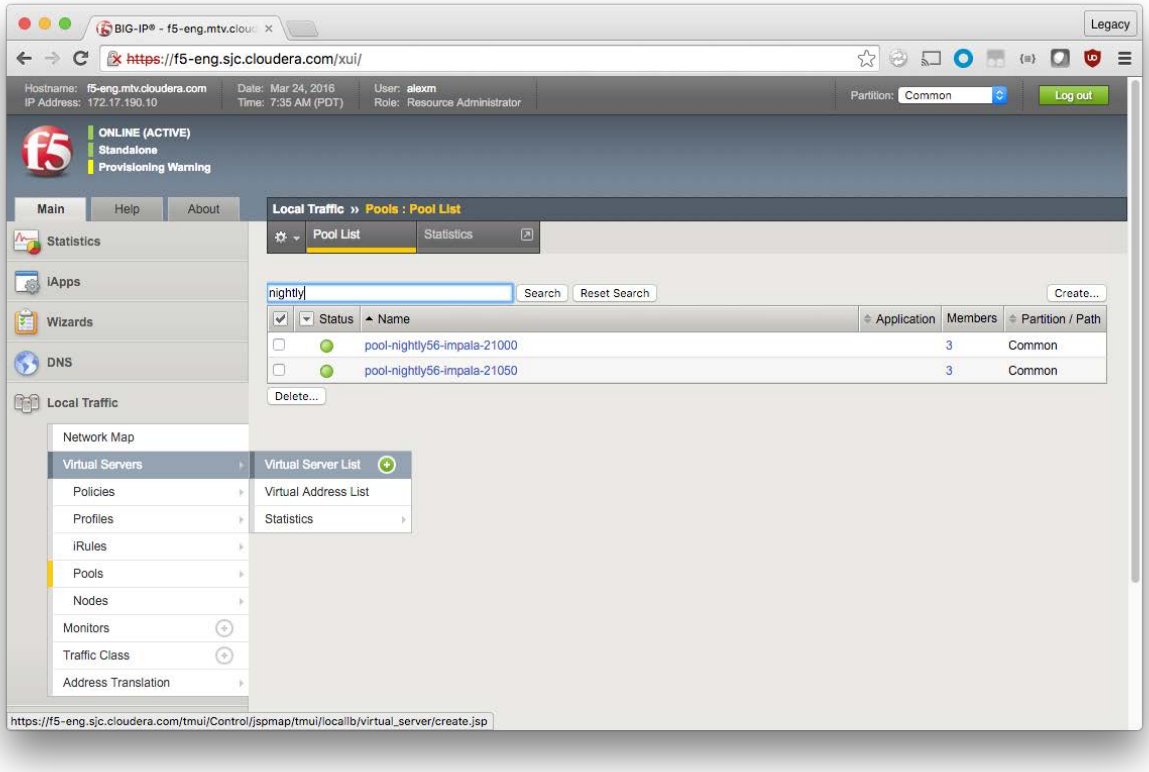


Create the Virtual Servers

A Virtual Server is the client-facing side of the load balancer—the IP and port that the client connects to for a particular service. Virtual Servers are backed by one or more pools of backend nodes; in most cases, the client is unaware of the backend nodes.

Port 21000

Local Traffic > Virtual Servers > Virtual Server List > Create (green plus)



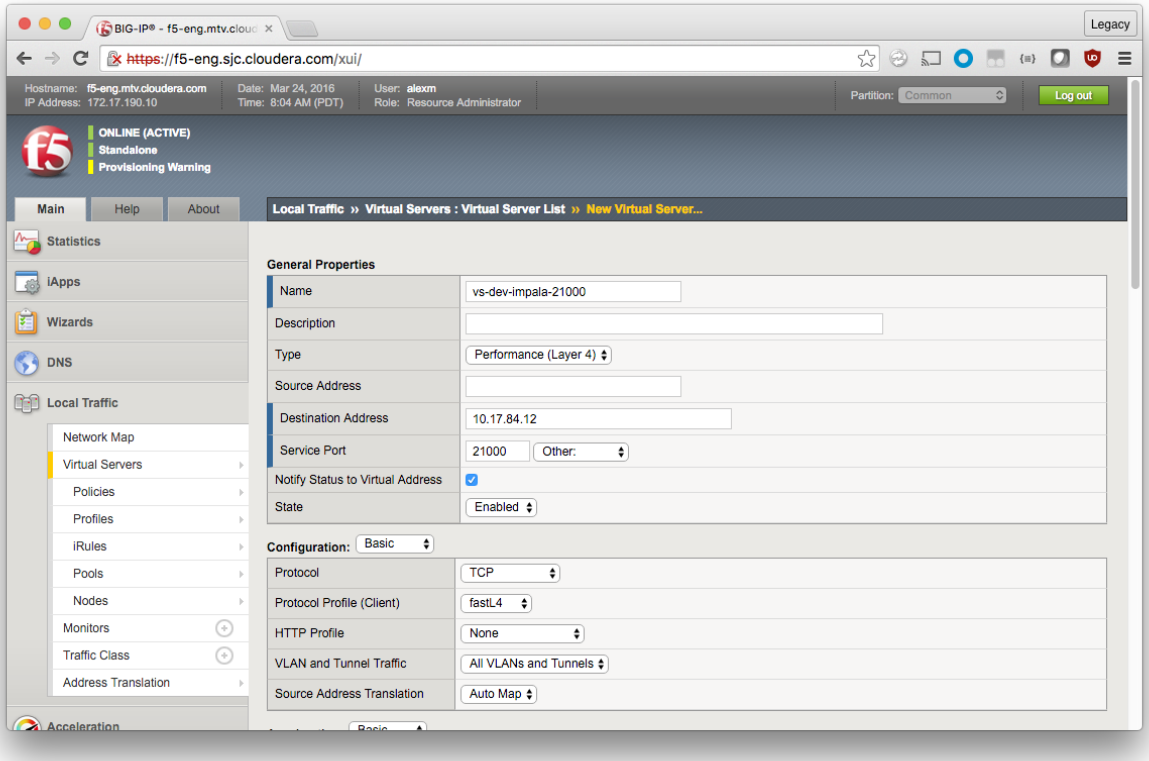
Here, you set all the required properties (denoted with blue bars to the left of the field name) as well as some optional properties.

General Properties

Name: vs-dev-impala-21000
Type: Performance (Layer 4)
Destination Address: <Self IP>
Service Port: 21000

The Self IP is the address that clients communicate with. Your FQDN should resolve to this address.

You can change the service port, but if you leave it as the default for Impala, you do not need to make changes to the clients.



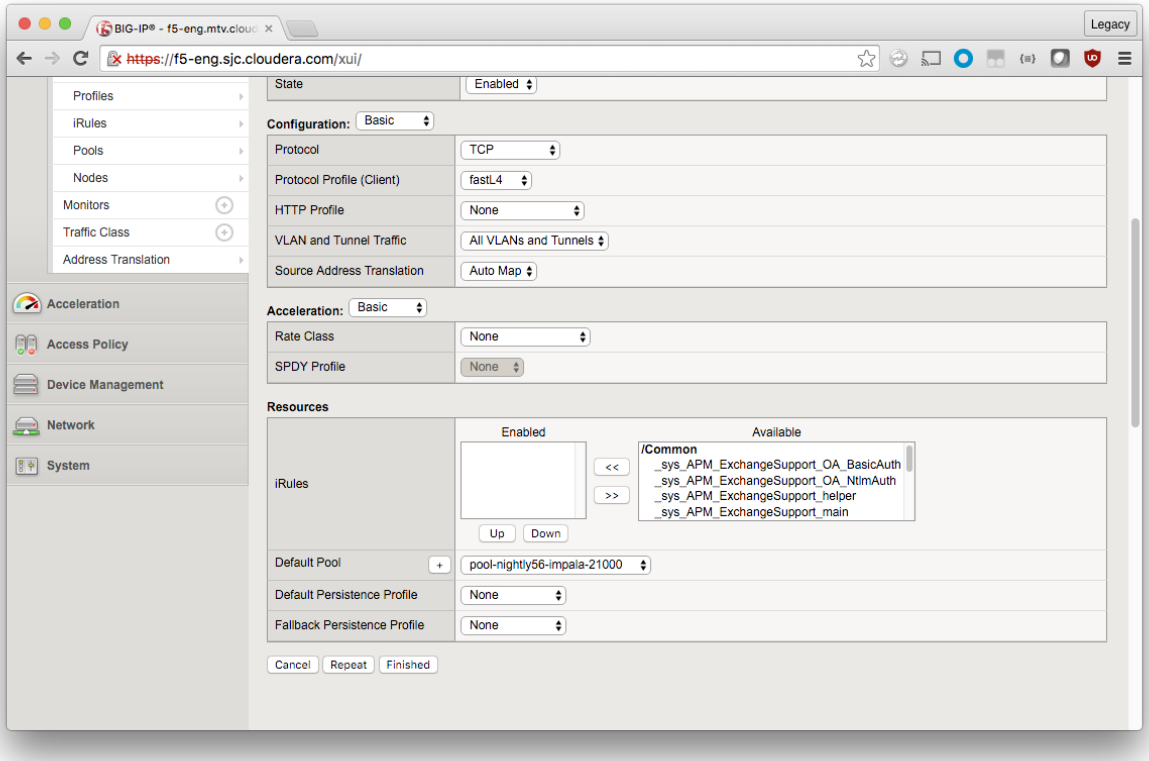
In addition, you must configure Source Address Translation and the default pool.

Configuration (Basic)

Source Address Translation: Auto Map

Resources

Default Pool: pool-nightly56-impala-21000



3. Click **Finished**.

Port 21050

Local Traffic > Virtual Servers > Virtual Server List > Create (green plus)

General Properties

Name: vs-dev-impala-21050
Type: Performance (Layer 4)
Destination Address: <Self IP>
Service Port: 21050

Configuration (Basic)

Source Address Translation: Auto Map

Hue requires persistent (or "sticky") sessions, meaning its requests need to be serviced by the same node when possible. So in addition to the default pool, you also configure a Persistence Profile for this Virtual Server. Without persistent sessions, Hue can be disconnected from long-running queries.

If you do not use Hue, leave Default Persistence Profile set to None.

Resources

Default Pool: pool-nightly56-impala-21050

Default Persistence Profile: source_addr

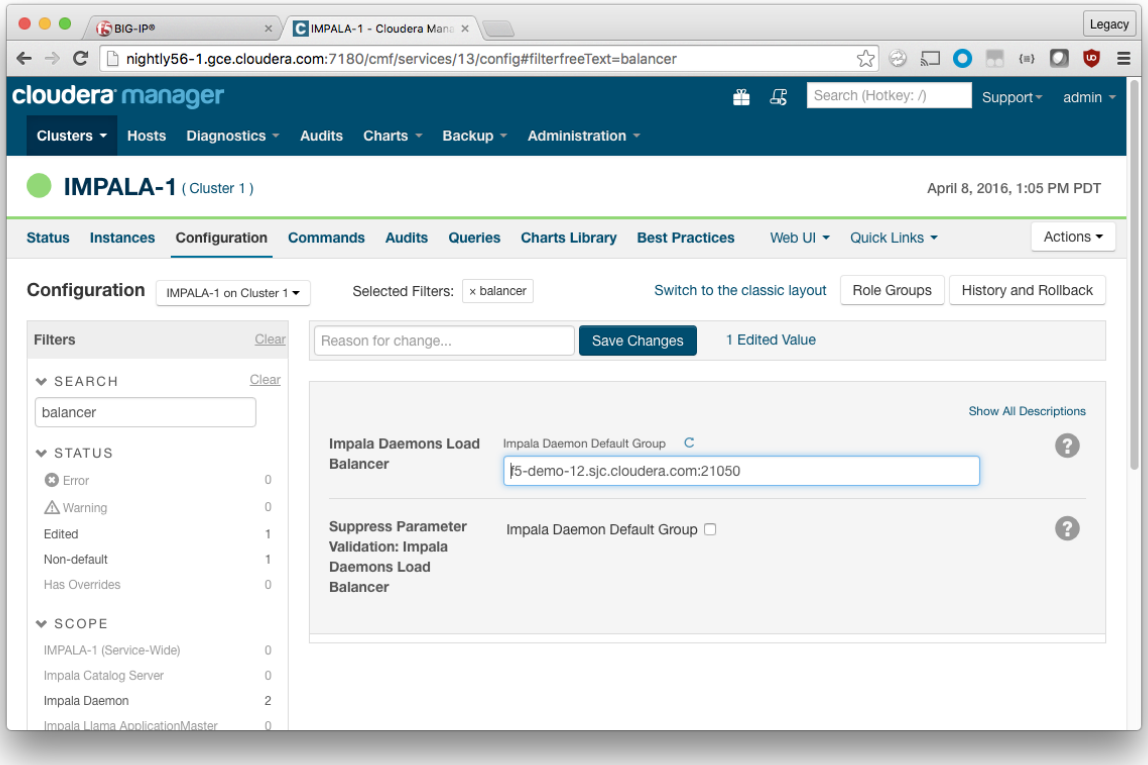
Click **Finished**.

Configure Impala

The following guidance mirrors the documentation for using a load balancer with Impala.

In Cloudera Manager, navigate to the Impala service, select the **Configuration** pane, then search for "balancer" to find the **Impala Daemons Load Balancer** parameter. The load balancer should be specified in *host*:*port* format, where host is your virtual server's FQDN and port. These values are used by Cloudera Manager and are also passed to Hue.

In the example, the self-IP FQDN is f5-demo-12.sjc.cloudera.com and the TCP port required is 21050. Type f5-demo-12.sjc.cloudera.com:21050 in the field.



If the **Impala Daemons Load Balancer** parameter is specified and Kerberos is enabled, Cloudera Manager adds a principal for 'impala/<load_balancer_host>@<realm>' to the keytab for all Impala daemons. No additional configuration is required for Kerberos.

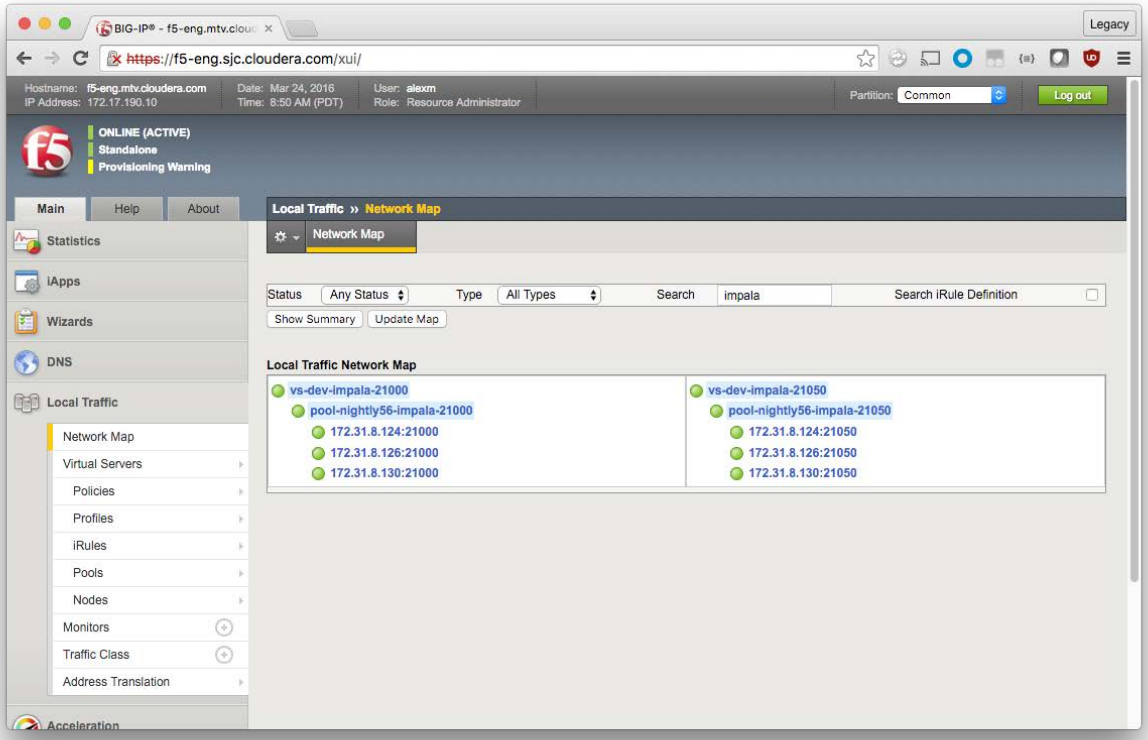
Click **Save**.

Note: If you are using Hue, you must restart the Hue service for the change to take effect.

Verification

Local Traffic Network Map

After everything is created, view the **Network Map** to see if everything is mapped correctly (**Local Traffic > Network Map**). You should see two virtual servers (on ports 21000 and 21050), each with a pool of backend Impala nodes. In this configuration, the port of the Virtual Server should match the ports of the pool nodes.



Throughout verification and operation, you can watch connection statistics in the F5 UI.

Statistics > Module Statistics > Local Traffic

Display Options
Statistics Type: Pools
Data Format: Normalized

If you have many pools, enter a search term to limit the number of pools displayed.

impala Search Reset Search

	Status	Pool/Member	Partition / Path	Bits In	Bits Out	Packets In	Packets Out	Connections Current	Connections Maximum	Connections Total
☐	●	pool-nightly56-impala-21000	Common	15.6K	32.7K	22	18	0	1	1
☐	●	-- nightly56-2:21000	Common	0	0	0	0	0	0	0
☐	●	-- nightly56-3:21000	Common	15.6K	32.7K	22	18	0	1	1
☐	●	-- nightly56-4:21000	Common	0	0	0	0	0	0	0
☐	●	pool-nightly56-impala-21050	Common	187.1K	753.6K	258	190	1	5	5
☐	●	-- nightly56-2:21050	Common	7.5K	1.7K	10	4	0	2	2
☐	●	-- nightly56-3:21050	Common	3.7K	896	5	2	0	1	1
☐	●	-- nightly56-4:21050	Common	175.8K	750.9K	243	184	1	2	2

Reset

JDBC

You can verify JDBC operation using IntelliJ and ImpalaJDBC (or other client) by connecting to jdbc:impala://f5-demo-12.sjc.cloudera.com:21050/ and running queries.

ODBC

ODBC can be verified by a client that utilizes the Simba Impala ODBC driver.

Impala Shell

Launch three terminal sessions using screen or tmux, and then launch impala-shell in each:

```
> impala-shell -i f5-demo-12.sjc.cloudera.com
Starting Impala Shell without Kerberos authentication
Connected to 10.17.84.12:21000
Server version: impalad version 2.4.0-cdh5.6.x RELEASE (build
85c0772d5455fe4ee5fe1d5fa39d162ad3c9e52f)
***********************************************************************
Welcome to the Impala shell. Copyright (c) 2015 Cloudera, Inc. All rights reserved.
(Impala Shell v2.4.0-cdh5.6.x (85c0772) built on Mon Mar 21 07:08:54 PDT 2016)

Run the PROFILE command after a query has finished to see a comprehensive summary
of all the performance and diagnostic information that Impala gathered for that
query. Be warned, it can be very long!
***********************************************************************

[f5-demo-12.sjc.cloudera.com:21000] > show tables;
Query: show tables
+-----------+
| name      |
+-----------+
| customers |
| sample_07 |
| sample_08 |
+-----------+
Fetched 3 row(s) in 0.50s
[f5-demo-12.sjc.cloudera.com:21000] > quit;
```

You should see one client connected to each pool member.

Hue

Hue inherits the **Impala Daemons Load Balancer** setting when set within Cloudera Manager (as described above). No other configuration is required, although Hue must be restarted for the setting to take effect.

Hue makes a connection and uses it for a period of time. It communicates to the impalad to determine the real host (the real hostname is visible on the Session tab of the Impala query page). After a few minutes of inactivity, LTM reaps the connection; when a fresh query is requested, Hue makes a new connection.

If you enabled persistence, you should see subsequent Hue query requests (and incrementing traffic) for a single pool.

Supplemental Configuration

Now that you are familiar with the basics, here are other configuration options that may be relevant to your installation.

Increasing Idle Timeouts

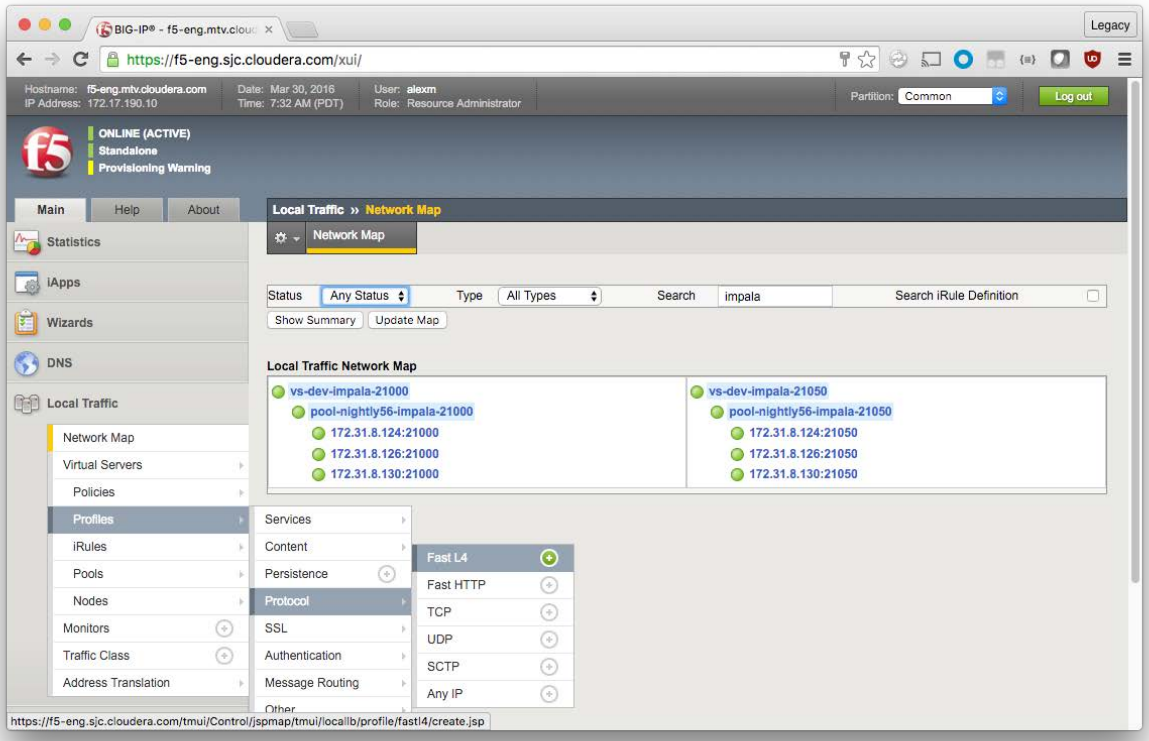
By default the LTM reaps idle connections after five minutes and persistence is limited to three minutes. For long-running queries, that might not be ideal. The following procedures set the effective idle timeout to one hour. In general, you set the value to the maximum query duration of the workload.

Keep in mind that network resources on the LTM are shared, and that Impala requirements must be weighed against the requirements of other LTM users.

Create a Custom Protocol Profile

By default, a Performance (Layer 4) Virtual Server uses the fastL4 protocol profile. If you just configure that profile to have a longer idle timeout, all other users of the profile are also affected. Instead, you create a new profile based on fastL4.

Local Traffic > Profiles > Protocol > Fast L4 > Create (green plus)



General Properties

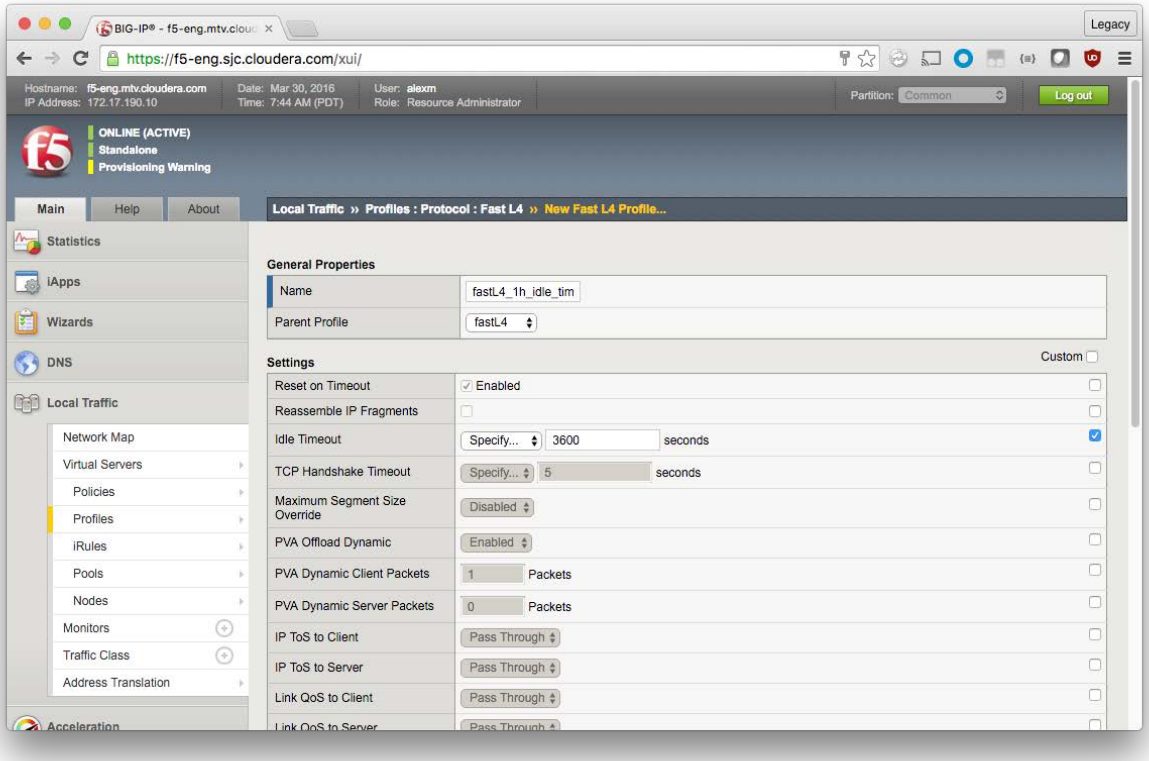
Name: fastL4_1h_idle_timeout

Parent Profile: fastL4

Because new profiles inherit all the properties of the parent profile, you should avoid changing the configuration on the default profiles. For each setting that you want to override, check the box to the right and make modifications.

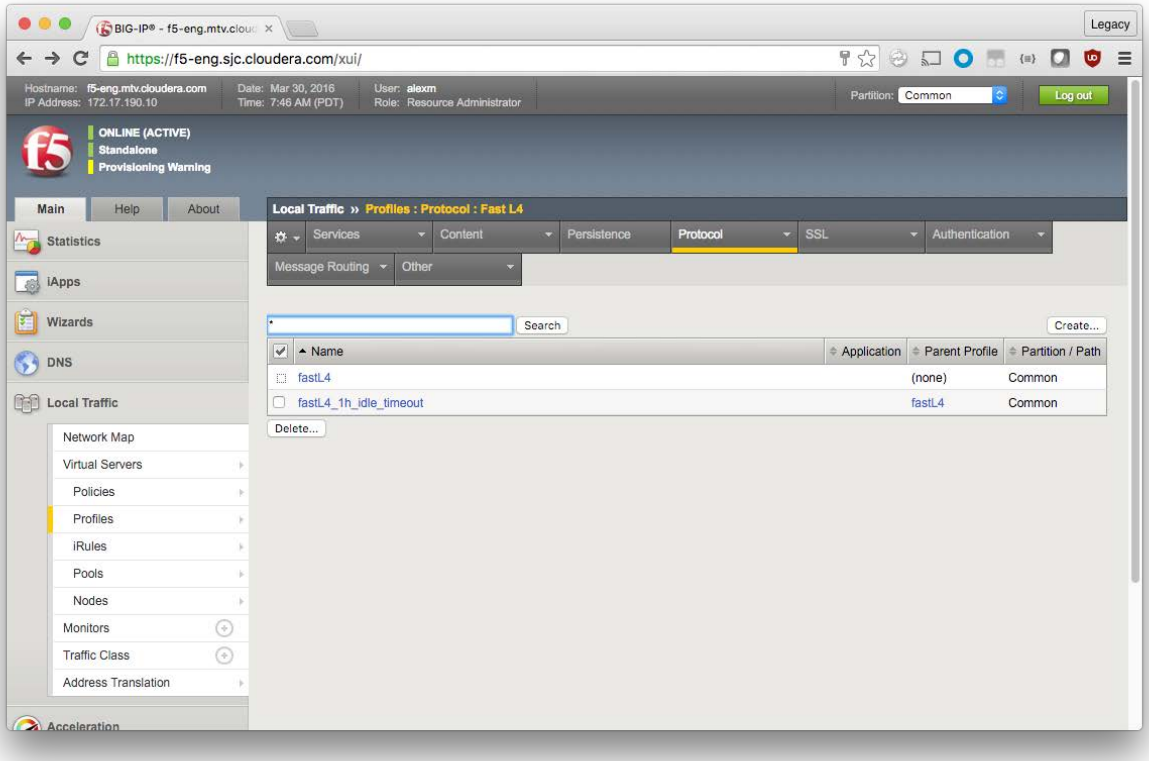
Settings

Idle Timeout: Specify, 3600



Scroll down and click **Finished**.

The new profile should appear.

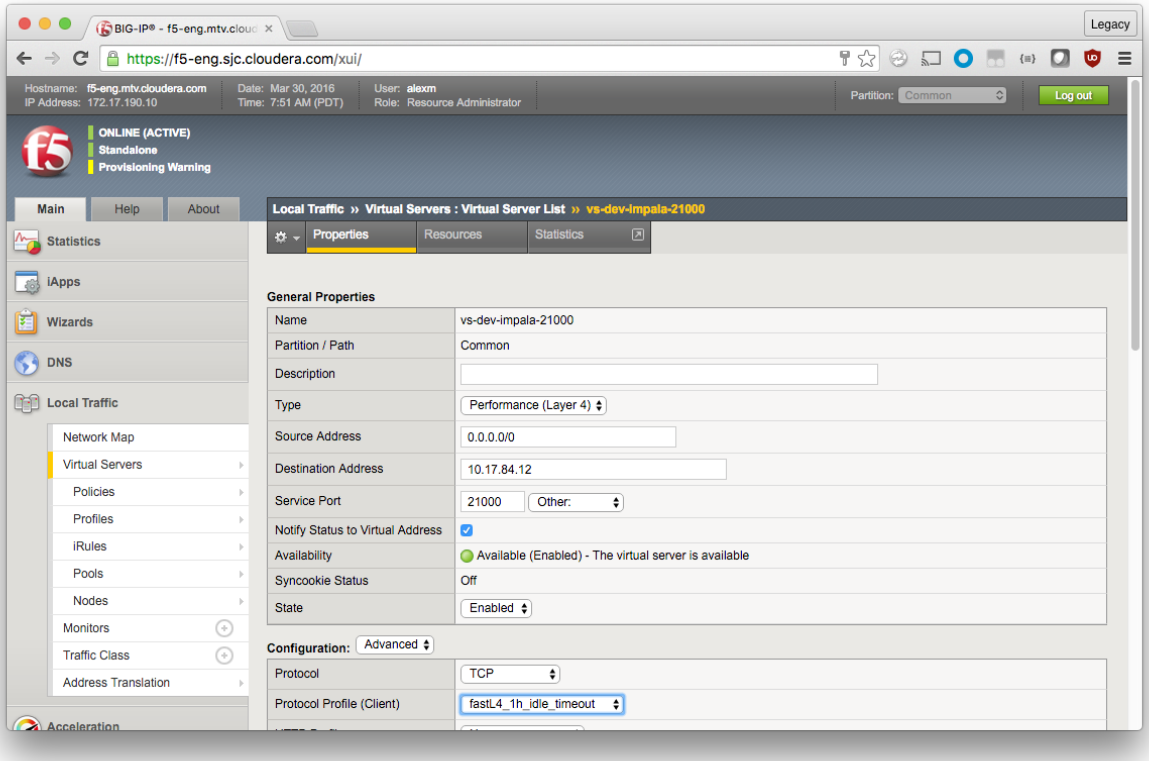


Apply a Custom Protocol Profile

Once the custom protocol profile has been created, configure the Virtual Server to use it instead of the default. Navigate to the Virtual Server you want to modify, and then make the following change. You must apply the profile to both port 21000 and 21050 Virtual Servers.

Configuration (Advanced)

Protocol Profile (Client): fastL4_1h_idle_timeout



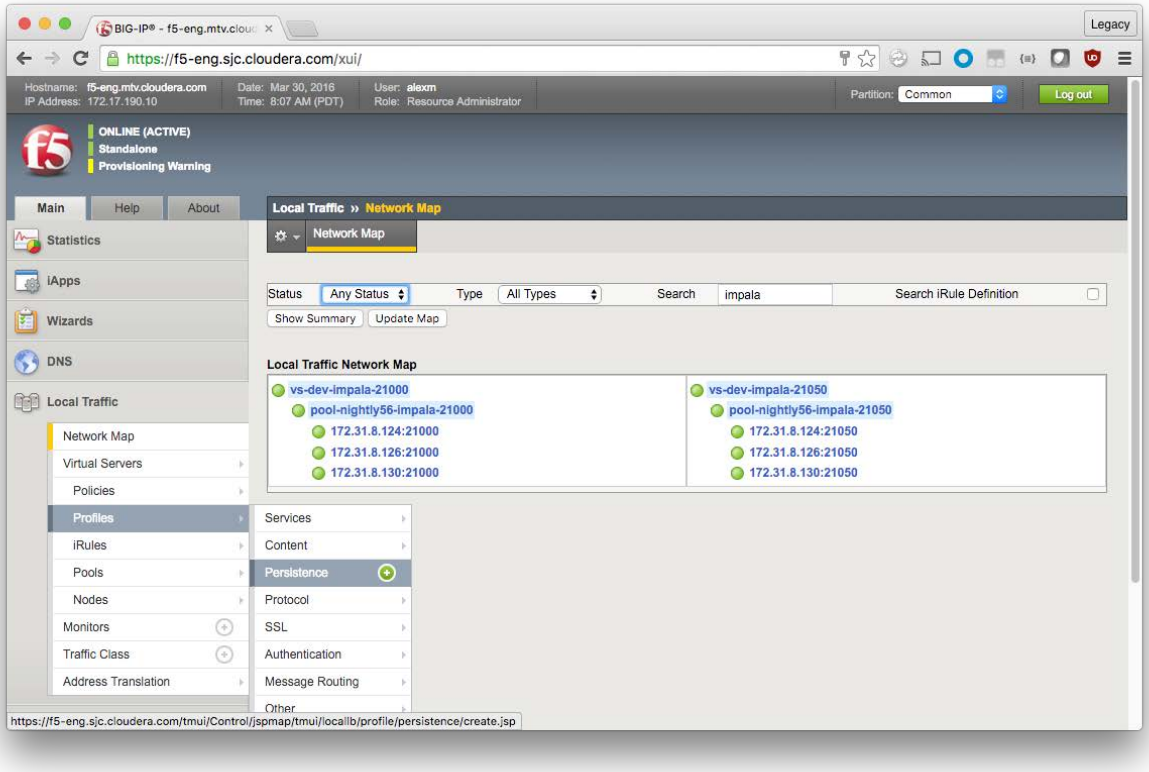
Scroll down and click **Update**.

Repeat for each Virtual Server you want to have an extended idle timeout.

Create a Custom Persistence Profile

Create custom persistence profiles in the same way as protocol profiles.

Local Traffic > Profiles > Persistence > Create (green plus)



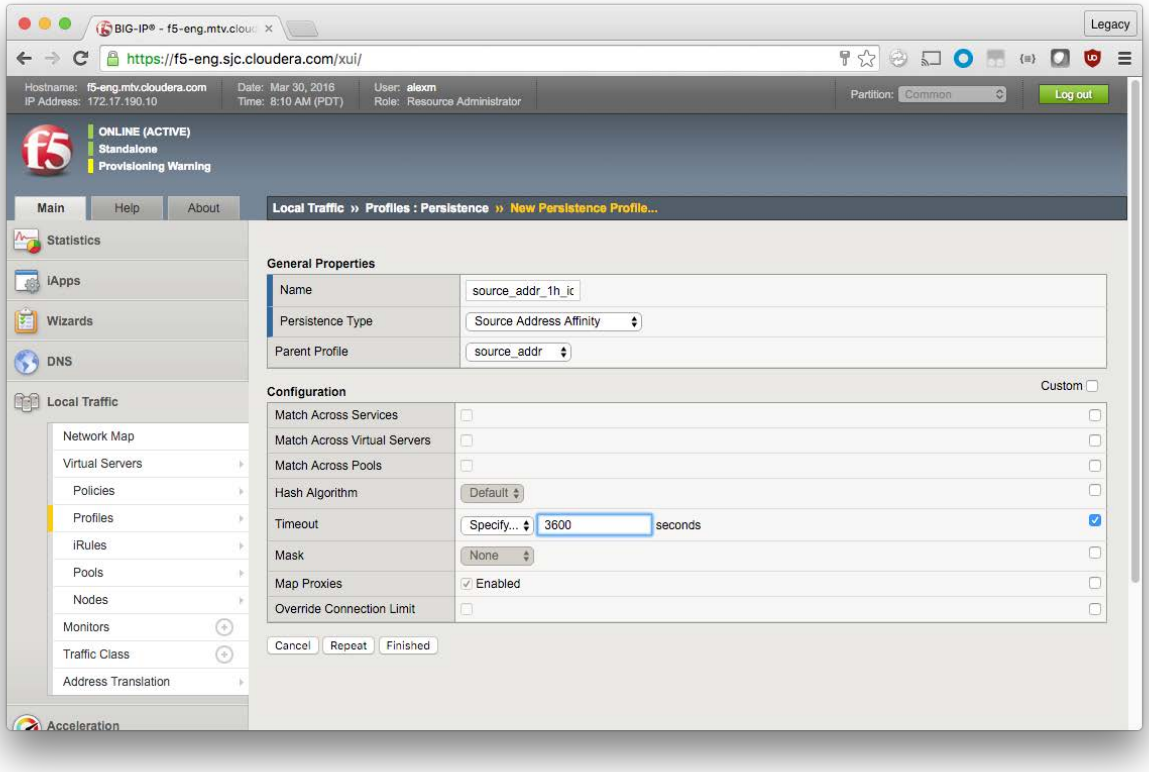
General Properties

Name: source_addr_1h_idle_timeout
Persistence Type: Source Address Affinity
Parent Profile: source_addr

Because new profiles inherit all the properties of the parent profile, you should avoid changing the configuration on the default profiles. For each setting that you want to override, check the box to the right and make modifications.

Settings

Timeout: Specify, 3600



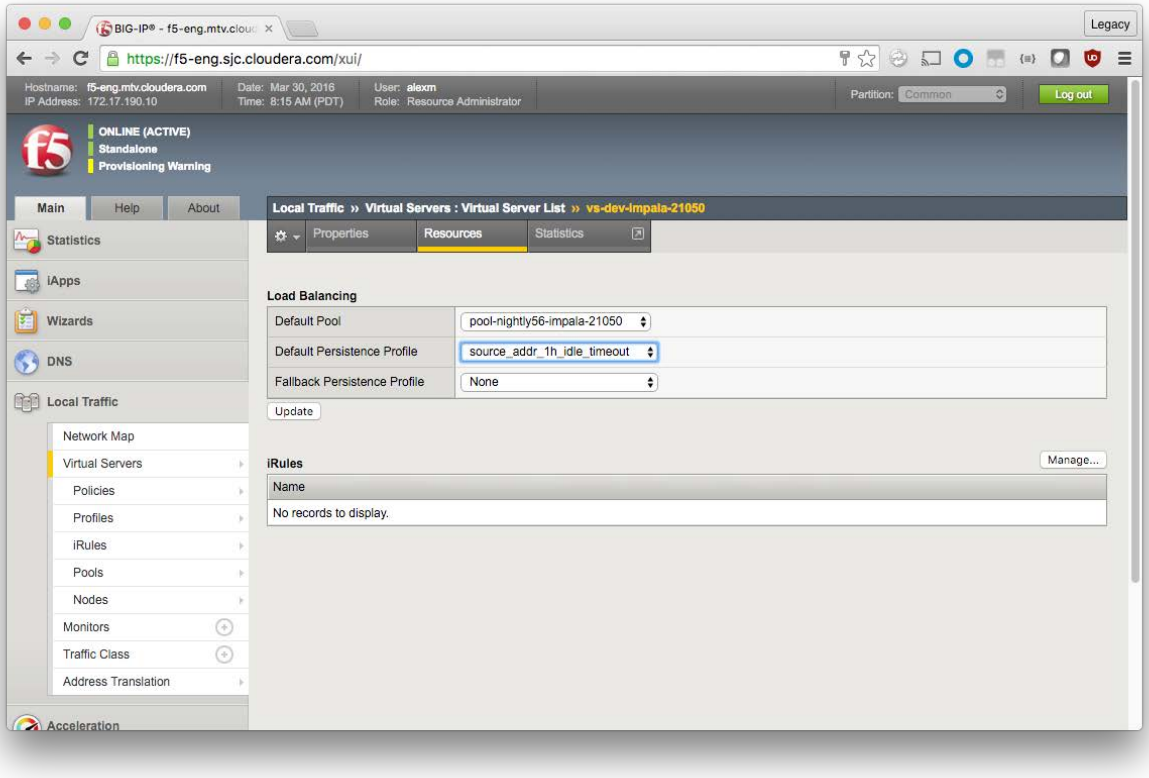
Click **Finished**.

Apply a Custom Persistence Profile

Navigate to the Virtual Server for Impala port 21050 (in the example, vs-nightly56-impala-21050). Click the **Resources** tab and make the following change.

Load Balancing

Default Persistence Profile: source_addr_1h_idle_timeout



Click **Update**.

Kerberos

If you followed Impala configuration instructions, no additional configuration is required for Kerberos.

TLS/SSL

When TLS/SSL is enabled for Impala, the client application—whether impala-shell, Hue, or something else—expects the certificate common name (CN) to match the hostname that it connected to. With no load balancer, the hostname and certificate CN are both that of the impalad instance. However, with a load balancer, the certificate presented by the impalad instance does not match the load balancer "front-end" hostname.

If you try to load-balance a TLS/SSL-enabled Impala installation without additional configuration, you see the following error when a client attempts to connect to the load balancer hostname:

```
Hostname we connected to "f5-demo-12.sjc.cloudera.com" doesn't match
certificate provided commonName "nightly57-kerberized-4.gce.cloudera.com"
(code THRIFTTRANSPORT): TTransportException(u'Hostname we connected to "f5-
demo-12.sjc.cloudera.com" doesn\'t match certificate provided commonName
"nightly57-kerberized-4.gce.cloudera.com"',)
```

You can configure an LTM in several ways to load-balance Impala:

- Client/Server SSL
- TLS/SSL Passthrough
- TLS/SSL Offload

Client/Server SSL

In this configuration, the LTM presents an SSL certificate to the client, decrypts the client request, then reencrypts the request before sending it to a backend impalad instance in the pool. This is often referred to as client/server SSL. At no point is the request or resulting payload unencrypted in transit. The client and server certificates can be managed separately, which can be convenient.

Modify the configuration process by applying a Client and Server profile. To quote the BIG-IP LTM manual on Managing SSL Traffic:

> A **Client profile** is a type of traffic profile that enables the BIG-IP system to accept and terminate any client requests that are sent by way of a fully SSL-encapsulated protocol. A **Server profile** is a type of profile that enables the BIG-IP system to initiate secure connections to a target web server.

To configure, perform the following tasks from the BIG-IP LTM documentation:

1. Install a key/certificate pair on the BIG-IP for terminating client-side secure connections.
2. Configure a client SSL profile (use the default server SSL profile).
3. Associate the profile with a virtual server.

These steps assume the pools and virtual servers have already been created following earlier guidance and that Impala has been configured for TLS/SSL.

Create/Import Certification and Key

Follow the BIG-IP documentation to create or import an SSL certificate and corresponding key. The certificate common name (CN) should match the load balancer hostname that you will use for Impala traffic.

If your certificate is self-signed, append the certificate to the Impala TLS/SSL CA Certificate file. You can find the file in Cloudera Manager by searching for **ssl_client_ca_certificate** in the Impala service configuration. The append is straightforward:

```
impalad-host$ cat certificate.crt >> /etc/certs/truststore.pem
```

If your certificate is signed by a CA already listed in the file, no further action is required. If not, append the client CA certificate to the file.

Create a Client SSL Profile

Local Traffic > Profiles > SSL > Client > Create (green plus)

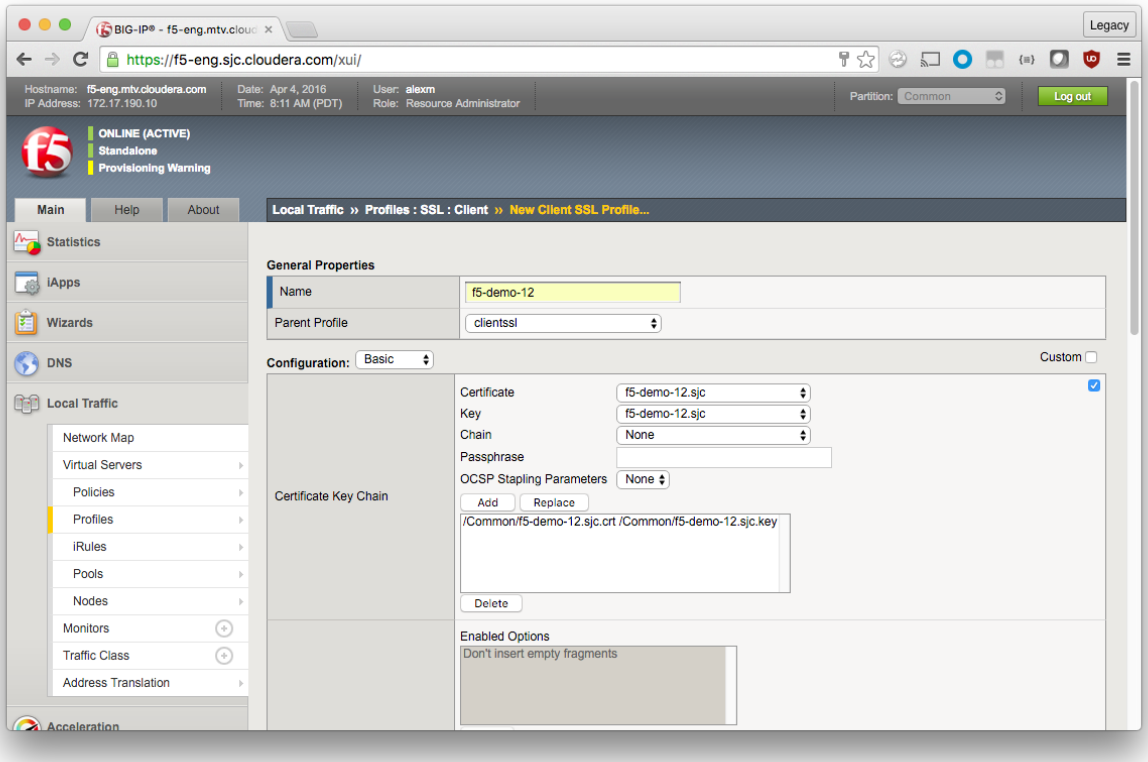
Name: f5-demo-12
Parent Profile: clientssl

Configuration (Basic)
Click the small blue checkbox to the right, and then complete the following fields:

Certificate: f5-demo-12.sjc
Key: f5-demo-12.sjc
Chain: None

In the example, my certificate (**f5-demo-12.sjc** in the BIG-IP SSL Certificate List) is self-signed, so there is no certificate chain. If you have a certificate chain or passphrase, make the appropriate selections.

Click **Add**.



Scroll down and click **Finished**.

Modify the Virtual Servers

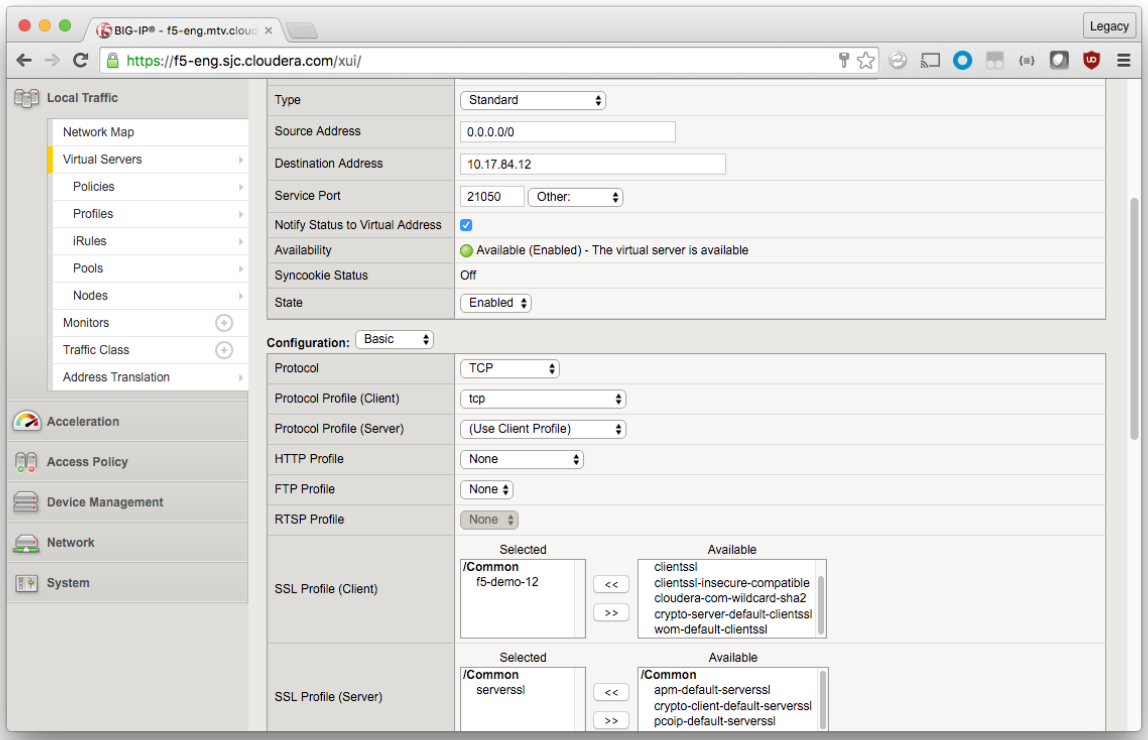
To use SSL profiles, you cannot use the Performance (Layer 4) type. You must change your existing Virtual Servers to use the Standard type and the SSL profile you just created.

General Properties
Type: Standard

Configuration: Basic

SSL Profile (Client): f5-demo-12

SSL Profile (Server): serverssl



Scroll down and click **Update**.

TLS/SSL Passthrough

In this configuration, TLS/SSL is terminated on the backend Impala instances; traffic is still encrypted end-to-end. No client-side SSL work is done by the LTM, and the encrypt/decrypt load on the backend hosts is not reduced.

1. Configure Impala with TLS/SSL.
2. Issue impalad certificates with a Subject Alternate Name (SAN) matching the frontend load balancer. See the example below.
3. Create pools and virtual servers as usual.
4. Configure Impala in Cloudera Manager as usual.

For example, if an impalad instance hostname is datanode05.sjc.cloudera.com, and the load balancer Virtual Server hostname is f5-impala.sjc.cloudera.com:

CN: datanode05.sjc.cloudera.com
SAN: datanode05.sjc.cloudera.com f5-impala.sjc.cloudera.com

Note: If the load balancer hostname is changed, *all* impalad certificates must be regenerated and redistributed, and the cluster must be restarted.

TLS/SSL Offload

In this configuration, TLS/SSL is terminated at the load balancer, and traffic between the backend Impala instances is unencrypted. Less overhead is incurred by the backend hosts because they do not have to encrypt or decrypt Impala traffic. This configuration presumes that cluster hosts reside on a trusted network and only external client-facing communication need to be encrypted in-transit. Traffic between Hue and the load balancer is also encrypted, however.

To configure:

1. Configure Impala without TLS/SSL.
2. Create pools.
3. Create/import an SSL certificate and key.
4. Create a Client SSL profile using the certificate and key.
5. Create a Standard Virtual Server with Client SSL profile; do not configure a Server SSL profile.
6. Configure Impala in Cloudera Manager as usual.
7. Configure Hue to use SSL for Impala: in Cloudera Manager, add the following text in the **Hue Service Advanced Configuration Snippet (Safety Valve) for hue_safety_valve.ini** field (substituting the path to your truststore).

```
[impala]
[[ssl]]
enabled=true
validate=false
cacerts=/etc/certs/truststore.pem
```

8. If your SSL certificate is signed by a CA already listed in the **cacerts** truststore, no further action is required. If not, append the CA certificate to the truststore file (or the certificate itself if self-signed).

You must restart Impala and Hue for the changes to take effect.

Verification

To verify TLS/SSL + Kerberos configuration, you can use Hue or impala-shell. You kinit before launching impala-shell:

```
> impala-shell -k --ssl -i f5-demo-12.sjc.cloudera.com
Starting Impala Shell using Kerberos authentication
Using service name 'impala'
SSL is enabled. Impala server certificates will NOT be verified (set --ca_cert to
change)
Connected to f5-demo-12.sjc.cloudera.com:21000
Server version: impalad version 2.4.0-cdh5.6.x RELEASE (build
85c0772d5455fe4ee5fe1d5fa39d162ad3c9e52f)
***********************************************************************************
Welcome to the Impala shell. Copyright (c) 2015 Cloudera, Inc. All rights reserved.
(Impala Shell v2.4.0-cdh5.6.x (85c0772) built on Mon Mar 21 07:08:54 PDT 2016)
```

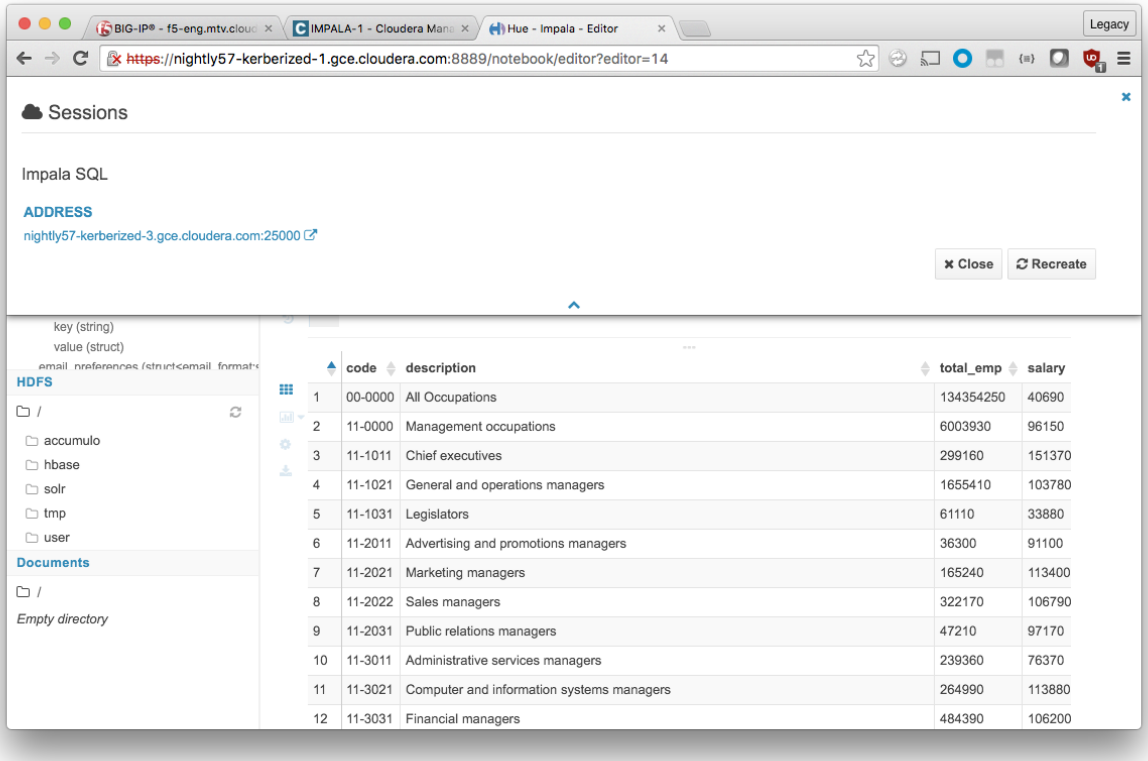
```
Run the PROFILE command after a query has finished to see a comprehensive summary
of all the performance and diagnostic information that Impala gathered for that
query. Be warned, it can be very long!
************************************************************************************

[f5-demo-12.sjc.cloudera.com:21000] > show tables;
Query: show tables
+-----------+
| name      |
+-----------+
| customers |
| sample_07 |
| sample_08 |
+-----------+
Fetched 3 row(s) in 0.50s
[f5-demo-12.sjc.cloudera.com:21000] > quit;
```

Known Issues

Backend nodes visible in Hue

Even when configured to utilize a load balancer, Impala clients are often aware of the backend impalad instance they are connected to. These hostnames and ports are visible in the Hue Sessions display and other places. This is normal and does not affect operation.



Error 104: Connection reset by peer

During normal operation, an Impala client keeps a user session open to the backend impalad instance. If an impalad instance becomes unavailable unexpectedly, the Impala connection is lost.

Without a load balancer, you need to know the hostname of another impalad instance; changes to the client application may be required to reconnect to the new instance.

With a load balancer, no client reconfiguration or knowledge about other backend hosts is required, but you are not connected to a new impalad instance until a new request or query is made.

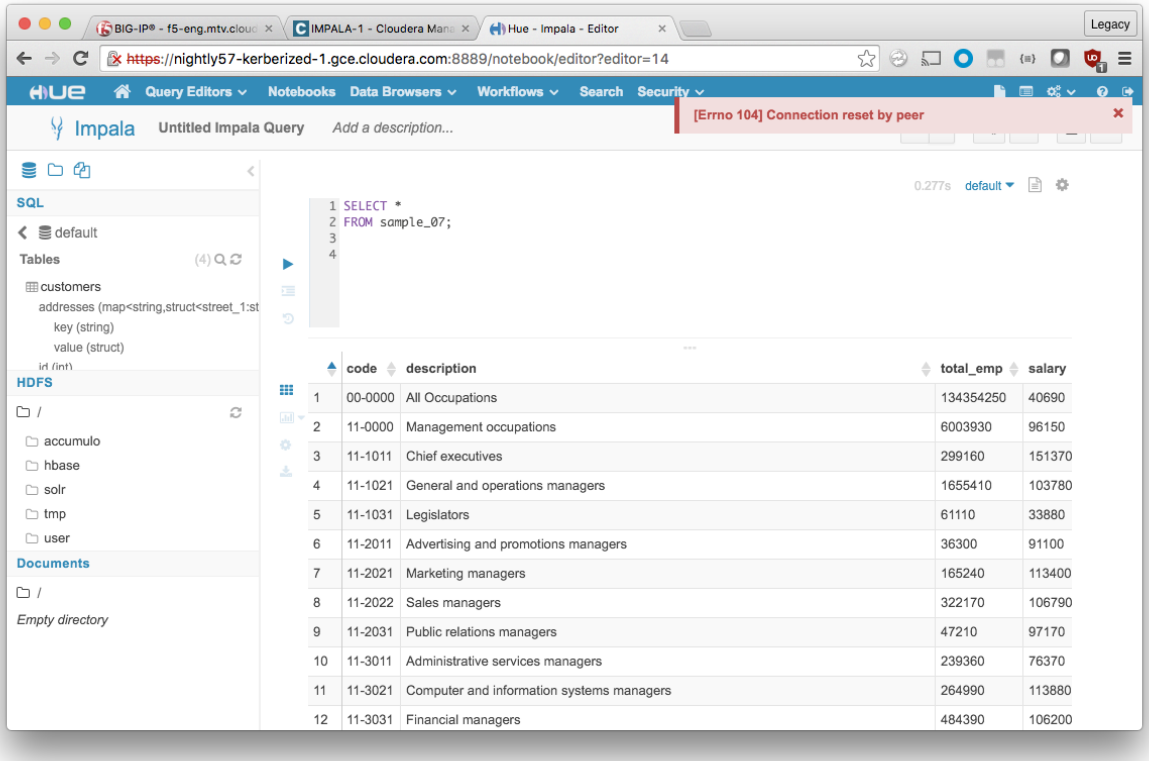
In impala-shell, you see the following:

```
[f5-demo-12.sjc.cloudera.com:21050] > show tables;
Connection lost, reconnecting...
Socket error 104: Connection reset by peer
```

To reestablish a session in impala-shell, CONNECT to the load balancer hostname and port.

```
[Not connected] > connect f5-demo-12.sjc.cloudera.com:21050;
Connected to f5-demo-12.sjc.cloudera.com:21050
Server version: impalad version 2.5.0-cdh5.7.1 RELEASE (build
f1464330fcc33b3709490a67a7ad1241ee983a3c)
[f5-demo-12.sjc.cloudera.com:21050] >
```

In Hue, the error looks like this:



In Hue, the your next query or Impala action results in a fresh connection being established with an operational impalad instance.

TTransportException, Could not start SASL

When TLS/SSL + Kerberos is enabled, you see the following error if you have not restarted Impala after configuring the load balancer host and port in Cloudera Manager. You see this even if you have a valid Kerberos ticket as seen through klist.

```
# impala-shell -k --ssl -i f5-demo-12.sjc.cloudera.com:21051
Starting Impala Shell using Kerberos authentication
Using service name 'impala'
SSL is enabled. Impala server certificates will NOT be verified (set --ca_cert
to change)
Error connecting: TTransportException, Could not start SASL: Error in
sasl_client_start (-1) SASL(-1): generic failure: GSSAPI Error: Unspecified
GSS failure.  Minor code may provide more information (Server
krbtgt/SJC.CLOUDERA.COM@GCE.CLOUDERA.COM not found in Kerberos database)
```

Hue

To spread the load across multiple Hue instances, you can configure Hue high availability.

References

Cloudera Documentation

- Using Impala through a Proxy for High Availability
- User a Load Balancer with Impala
- Ports Used by Impala

F5 Documentation

- Glossary and Terms
- Manual Chapter: Configuring Load Balancing Pools
- SOL14163: Overview of BIG-IP virtual server types (11.x)
- Manual Chapter: Session Persistence Profiles
- Manual Chapter: Managing SSL Traffic
- SOL14620: Managing SSL certificates for BIG-IP systems using the Configuration utility
- Technical Article: LTM: Action on Service Down