LTM Fundamentals v15.1

Participant Hands-on Lab Guide

Solutions for
an application world.

Last Updated: 11/20/2020

©2014 F5 Networks, Inc. All rights reserved. F5, F5 Networks, and the F5 logo are trademarks of F5 Networks, Inc. in the U.S. and in certain other countries. Other F5 trademarks are identified at f5.com.

Any other products, services, or company names referenced herein may be trademarks of their respective owners with no endorsement or affiliation, express or implied, claimed by F5.

TABLE OF CONTENTS

Table of Contents 3
BIG-IP® Local Traffic Manager (LTM) - V12 Lab Guide 5
Lab Overview 5
Scenario 6
Lab Network Diagram (based on Lamp4.0) 6
Accessing the Lab Environment 7
Access the Lab Environment ..... ***Error! Bookmark not defined.***
Lab 1: The Basics (Networking, Pools and Virtual Servers) 10
Creating VLANs *10*
Assigning a Self IP addresses to your VLANs *11*
Assigning the Default Gateway *12*
Creating Pools *13*
Creating Virtual Servers *14*
ExtraCredit! *16*
Lab 2: Load Balancing, Monitoring and Persistence 17
Ratio Load Balancing *17*
Priority Groups Lab *19*
Monitor Labs *21*
Content Monitors *22*
Persistence Labs *25*
Simple (Source Address) Persistence *25*
Cookie Persistence (Cookie Insert) *28*
Lab 3: Accelerating Applications Lab 30
TCP Express *30*
HTTP Optimization - RamCache Lab *31*
HTTP Optimization - HTTP Compression Lab *32*
Lab 4: SSL Offload and Security 33
Creating a Self-signed certificate and key *33*
Creating SSL Client Profile *34*
Building our New Secure Virtual Server *34*
Securing web applications with the HTTP profile *35*
Lab 5: BIG-IP Policies and iRules 36
Write an iRule to retrieve images when an HTTP request is received *36*
Use a BIG-IP Policy to retrieve images from a different pool *37*
Lab 6: Support and Troubleshooting 39

Archive the current configuration and perform a health check using a QKview *39*

Troubleshoot using TCPDump or Curl. *41*

Lab 7: Device Service Clusters (DSC) 43

Base Networking and HA VLAN *43*

Configure HA *44*

Bonus Lab – Traffic groups, iApps and Active-Active 46

Building a new traffic group and floating IP. *46*

Building an HTTP application using an iApp template. *46*

Active-Active Setup *47*

Appendix 48

BIG-IP Policy for retrieving jpeg images from the image_pool *48*

BIG-IP extra credit iRule to add PNG to access_image_pool iRule *48*

BIG-IP® LOCAL TRAFFIC MANAGER (LTM) - V15 LAB GUIDE

This lab guide is designed for you to get an understanding of the BIG-IP Local Traffic Manager (LTM) product.

Lab Overview

- F5 BIG-IP LTM VE, licensed using F5-BIG-VE-LAB-V18-LIC
- Your BIG-IP is as close to factory default as possible, only the following changes have been made:
    - The management IP has already been configured
    - The initial setup has been completed. (Licensing and Platform information)
    - The Idle Timeout was modified from 1200 seconds to 7200 seconds
    - The Welcome messages for the GUI and SSH were changed.
    - An archive file base-setup-and-licensing.ucs was created allowing you to revert to the base settings above.

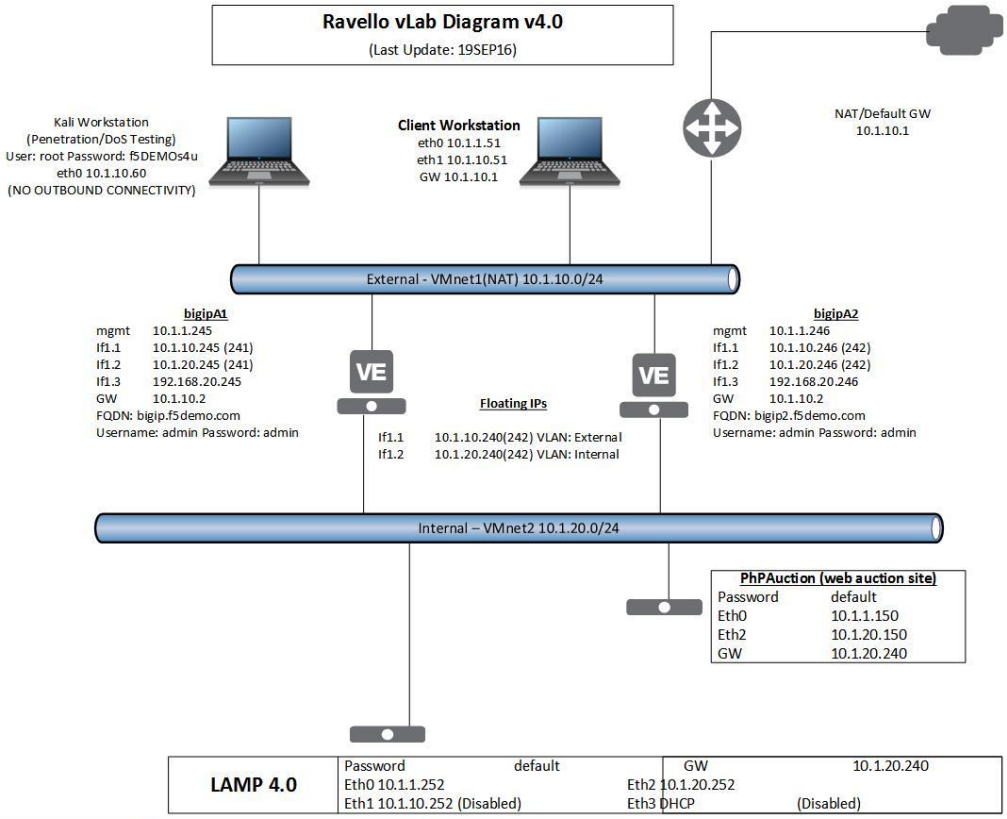
Various directory and application services are available within the lab environment.

DO NOT COPY INFORMATION FROM THE SCREENSHOTS. THEY ARE FOR REFERENCE ONLY.

Scenario

Your customer has the following environment. The servers sit on the customers internal VLANs, the virtual servers will exist in another VLAN in the DMZ. The customer does not want to rework their networking and does not wish to use the LTM as the default gateway. Our solution will be to use SNATs to force traffic to passing through the BIG-IP to return through the LTM.

LAB NETWORK DIAGRAM (BASED ON LAMP4.0)



LDAP

10.1.20.252:1389 or 389

People:
- user.0 - user.49
- adminuser
- corpuser
- remoteuser

Groups:
- employees
  - user.0 – user.10
  - corpuser
- remote
  - user.11 – user.20
  - remoteuser
- admin
  - adminuser
  - user.0
  - user.10
  - user.20
  - user.30

Password for all users:
- password

DNS (also see Reserved Virtual Ips)

10.1.20.252:53

dojo.f5demo.com.	10.1.10.50
idp1.f5demo.com.	10.1.10.100
app1.f5demo.com.	10.1.10.101
app2.f5demo.com.	10.1.10.102
idp2.f5demo.com.	10.1.10.103
idp3.f5demo.com.	10.1.10.104
cert.f5demo.com.	10.1.20.11
server1.f5demo.com.	10.1.20.11
server2.f5demo.com.	10.1.20.12
server3.f5demo.com.	10.1.20.13
server4.f5demo.com.	10.1.20.14
server5.f5demo.com.	10.1.20.15
app3.f5demo.com.	10.1.20.16
dvwa.f5demo.com.	10.1.20.17
peruggia.f5demo.com.	10.1.20.18
wackopicko.f5demo.com.	10.1.20.19
hackazone.f5demo.com.	10.1.20.20
websafe.f5demo.com.	10.1.20.50
auction.f5demo.com.	10.1.20.150
lamp.f5demo.com.	10.1.20.252
ldap.f5demo.com.	10.1.20.252
lamp2.f5demo.com.	10.1.20.252

Apache

Demo Servers

server1.f5demo.com	10.1.20.11:80,443,8081
server2.f5demo.com	10.1.20.12:80,443,8081
server3.f5demo.com	10.1.20.13:80,443,8081
server4.f5demo.com	10.1.20.14:80,443,8081
server5.f5demo.com	10.1.20.15:80,443,8081

Hackables

dvwa.f5demo.com	10.1.20.17	admin/password
peruggia.f5demo.com	10.1.20.18	admin/password
wackopicko.f5demo.com	10.1.20.19	admin/admin
hackazon.f5demo.com	10.1.20.20/admin	admin/admin
	Hackazone DB	root/default

FPS

websafe.f5demo.com	10.1.20.50 or 10.1.1.252	user/12345678
DemoTools	http://10.1.1.252/demotools.php	

Services

Webmin	localhost:10000	root/default
Splunk Web Interface	10.1.1.252:8000	admin/default
Splunk Logging Port	10.1.20.252/5514 (configured)	
SimpleSAMLPHP	localhost/simplesaml/	
OpenAM	localhost:8080/openam	
Sert.f5demo.com	server1.f5demo.com /CaPolicy/ /CaRevocation/ /Policy/ /Revocation/	

RADIUS

10.1.20.252:1812

Clients:
- 10.1.20.240
- 10.1.20.241

Secrets:
- f5networks (for clients)
- testing123 (for def tests)

Users:
- user0/password
- user1/password

Other

FTP/SCP	10.1.1.252 10.1.20.252 f5 or root / default
SYSLOG	port 514
NTP	port 123
TinyCA	cert.f5demo.com pwd: f5networks
Splunk (Xubntu)	http://localhost:8000 admin / default
VNC	10.1.1.252:5900 password: default

ACCESSING THE LAB ENVIRONMENT

Accessing the Lab Environment

To access F5's Unified Demo Framework (UDF) environment you will have to have an active account.

If you have UDF account and you have received a confirmation email from **noreply@registration.udf.f5.com** for the **201 Certification Lab - TMOS Administration**. You are ready to access the labs.

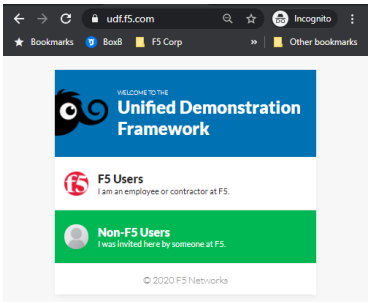
If you did not receive the confirmation please check your junk folder or simply try your credentials (you can only see labs you have been invited to). If you still cannot gain access to the course notify the instructor.

If you do NOT have an active account you should have received an email from **noreply@registration.udf.f5.com** with you username (email address) and a temporary password. If you did not receive an email please check your junk mail folder to see if the email was flagged as spam. Activate your account by setting your password and then proceed below.

Accessing the UDF labs

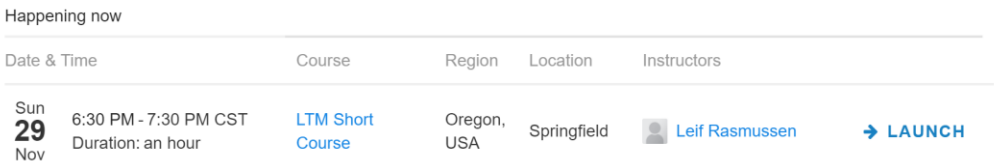
You will be access the labs using the F5 Unified Demo Framework (UDF). **Chrome** is the preferred browser for UDF access. Other browsers may work but are not supported.

1. Open your browser and navigate to the F5 UDF https://udf.f5.com
   - Select the **Non-F5 Users** option and log in using your UDF credentials.



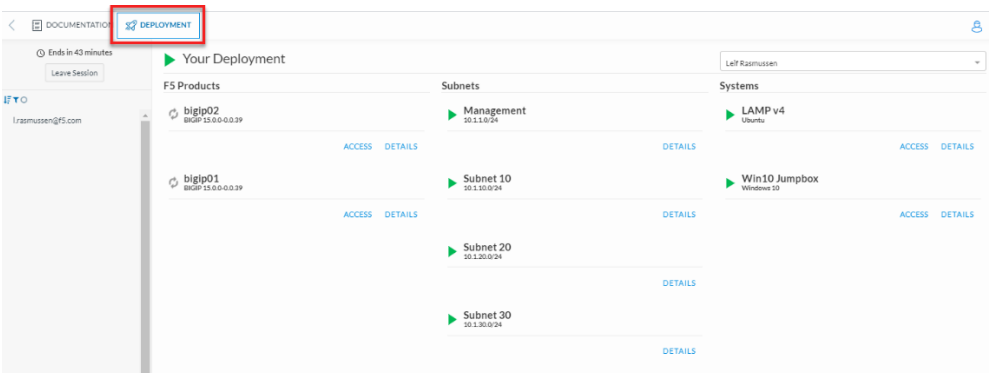
Important:
You should retain these credentials, as they will be required to any access future F5 UDF courses you attend in the F5 UDF environment

2. You should see the event(s) under **Happening now**. Find the **LTM Short Course** event and click on the **Launch** link at the far right.

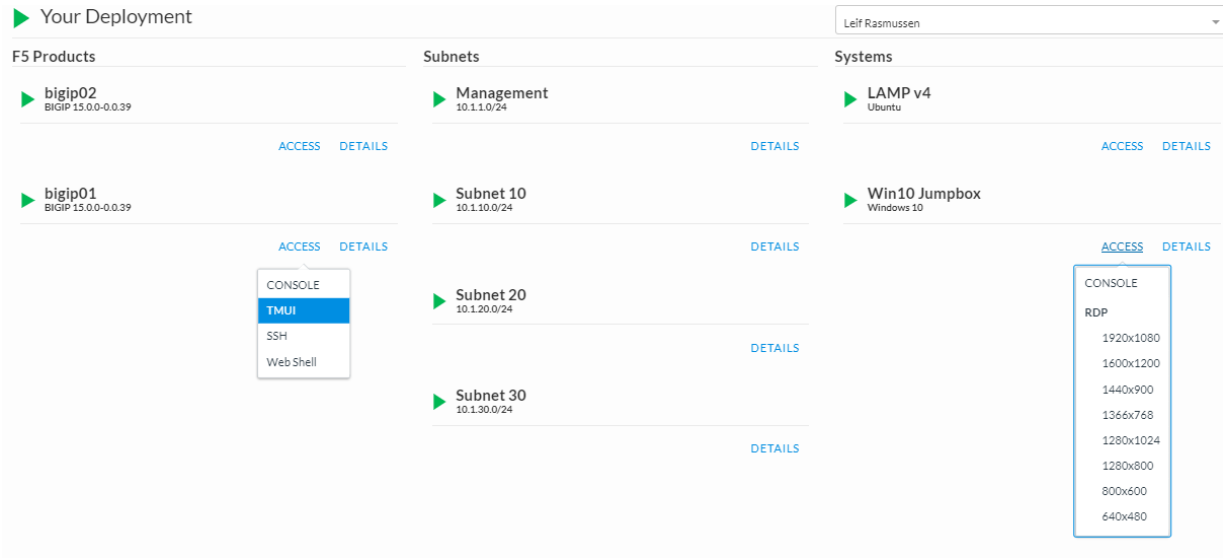


3. Click the **Join** button. ***Manage SSH Keys*** *should not be required.*
4. At the top you will see **Documentation** and **Deployment**. In the **Documentation** section you can elect to leave the session, see how long the session will last and course documentation.

5. Click on the **Deployment** tab. The VM instances will take a few minutes to provision and will be ready when you have a green arrow.



6. To access an instance, click the **Access** link and select the type of access you want from the drop-down menu.



Note
If you use the Web Shell use ctrl-shift-v to paste.

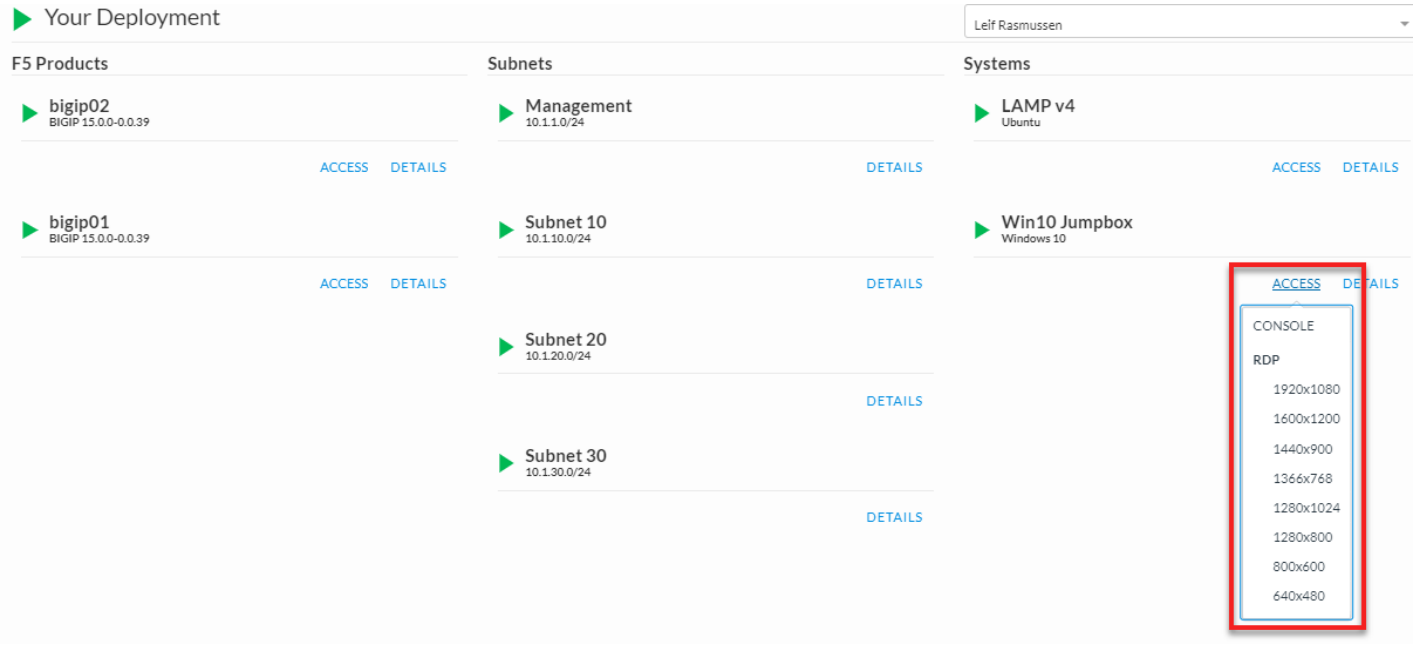
Lab Environment

Important

Components	Mgmt IP	Access	Username	Password
bigip01	10.1.1.4	GUI	admin	f5UDFrocks!
	10.1.1.4	SSH	root	f5UDFrocks!
bigip02	10.1.1.5	GUI	admin	f5UDFrocks!
	10.1.1.5	SSH	root	f5UDFrocks!
Win10 Jumpbox	10.1.1.6	RDP	f5student	f5UDFrocks!
Lamp v4	10.1.1.7	SSH	f5student	f5UDFrocks!
	10.1.1.7	webmin	f5student	f5UDFrocks!

Accessing the Win10 Jumpbox to begin the labs

In the **Deployments** tab and select the **Access** drop down menu and under **Win10 Jumpbox** select **RDP** and the screen size. Log on with the credentials in the table above.



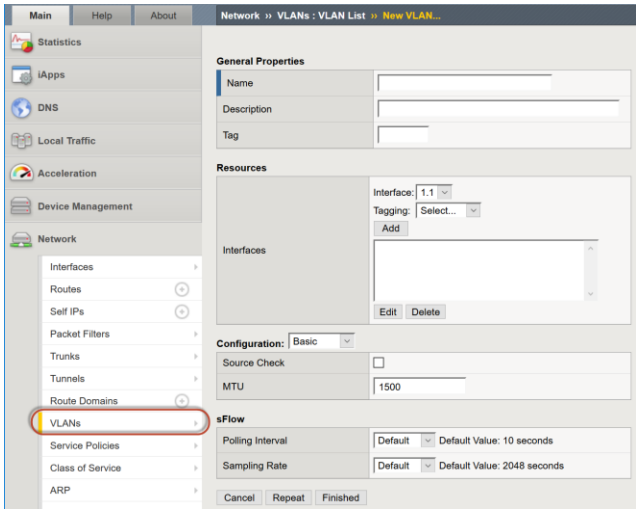
LAB 1: THE BASICS (NETWORKING, POOLS AND VIRTUAL SERVERS)

In this lab we will access the Management GUI. We will then create the VLANs, and assign a self IP addresses to our VLAN. As mentioned during our lecture portion, BIG-IPs may be put in-line or one-armed depending on your customer's requirements and topology.

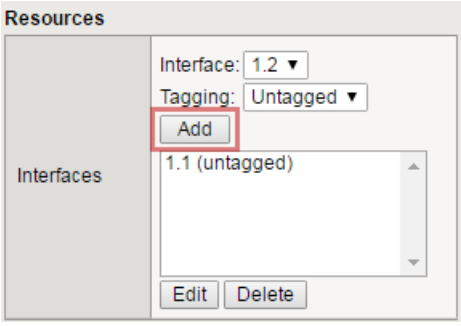
Creating VLANs

You will need create two untagged VLANs, one client-side VLAN (**client_vlan**) and one server-side VLAN (**server_vlan)** for the devices in your network.

1. From the sidebar select **Network > VLANs** then select **Create**



   a. Under **General Properties**:
      i. **Name**: client_vlan
   b. The name is for management purposes only, you could name them after your children or pets
      i. **Tag**: <leave blank>
         1. Entering a tag is only required for "**Tagged**" (802.1q) interfaces, "**Untagged**" interfaces will automatically get a tag which is used for internal L2 segmentation of traffic.
   c. Under **Resources** in the **Interfaces** section:
      i. **Interface**: 1.1
      ii. **Tagging**: Untagged
      iii. Select the **Add** button. Leave all other items at the default setting.

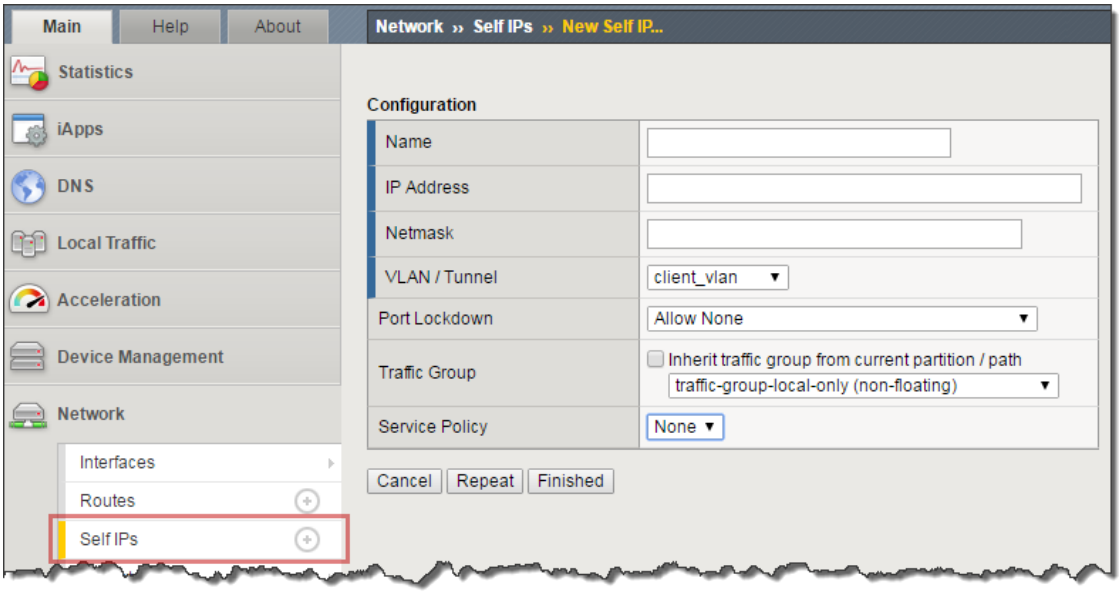


      iv. When you have completed your VLAN configuration, hit the **Finished** button

Create another untagged VLAN named **server_vlan** on interface **1.2.**

Assigning a Self IP addresses to your VLANs

1. Go to **Network > Self IPs**, select **Create**.

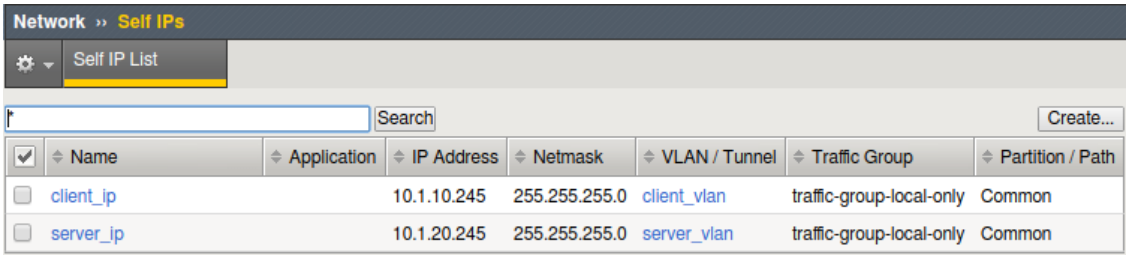


a. Create a new self IP, for the **server_vlan** and **client_vlan** VLANs. In **Network >> Self IPs >> New Self IP**, under **Configuration** enter:

		Server-Side	**Client-side**
i.	**Name**:	server_ip	client_ip
ii.	**IP Address**:	10.1.20.245	10.1.10.245
iii.	**Netmask**:	255.255.255.0	255.255.255.0
iv.	**VLAN**:	server_vlan	client_vlan
v.	**Port Lockdown**:	Allow None	Allow None

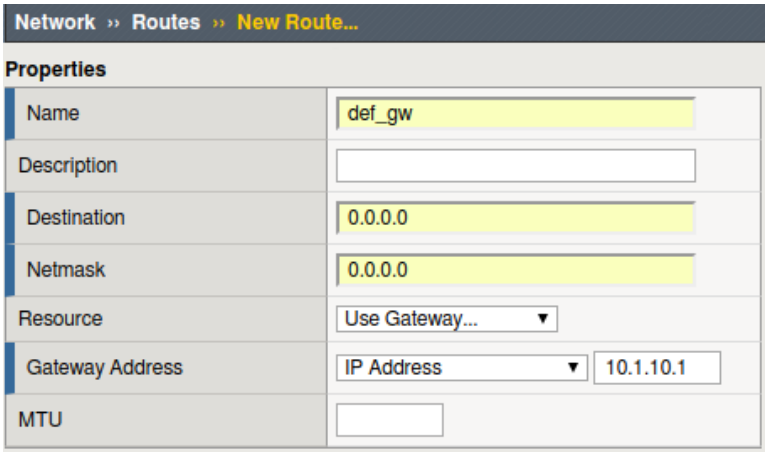
   1. The default "**Allow None**" means the Self IP would respond only to ICMP.
   2. The "**Allow Defaults**" selection opens the following on the self IP of the VLAN
      a. TCP: ssh, domain, snmp, https
      b. TCP: 4353, 6699 (for F5 protocols, such as HA and iQuery)
      c. UDP: 520, cap, domain, f5-iquery, snmp
      d. PROTOCOL: ospf

b. When you have completed your selfIP configuration, hit the Finished button. You should have something similar to the following:



Assigning the Default Gateway

1. Go to **Network > Routes** and then **Add**.
    a. Here is where we assign our default gateway (and other static routes as desired)



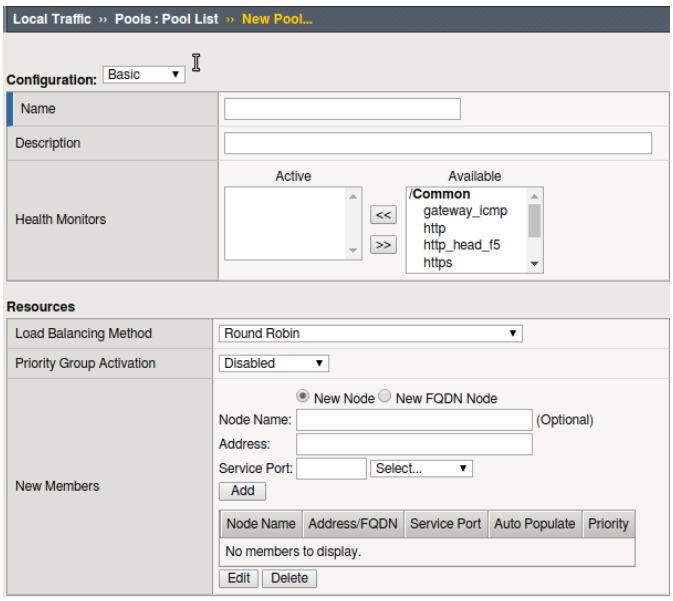
    b. Under **Properties**
        i. **Name**: def_gw
        ii. **Destination**: 0.0.0.0
        iii. **Netmask**: 0.0.0.0
        iv. **Resource**: Use Gateway…
        v. **Gateway Address**: 10.1.10.1
        vi. When you have completed defining your default gateway, hit the Finished button

2. Verify your network configuration
    a. Ping your client-side self ip (**10.1.10.245**) to verify connectivity
    b. Use an SSH utility, such as puTTY, to access your BIG-IP management port at 10.1.1.245.
        i. User: **root** Password: **f5UDFrocks!**
        ii. Ping your default gateway, 10.1.10.1
        iii. Ping a web server at 10.1.20.11.

Creating Pools

In this lab we will build a pool and virtual serve to support our web site and verify our configurations by accessing our web servers through the BIG-IP. Verification will be performed visually and through various statistical interfaces.

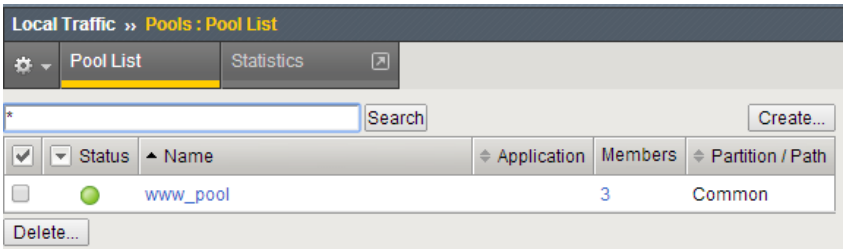
1. From the sidebar select **Local Traffic >> Pools** then select **Create**. Here we will create our new pool



   c. Under **Configuration**:
      i. **Name**: www_pool
         1. The name is for management purposes only, no spaces can be used
      ii. **Description**: <optional>
      iii. **Health Monitor**: http
   d. Under **Resources**
      i. **Load Balancing Method**: <leave at the default Round Robin>
      ii. **Priority Group Activation**: <leave at default>
      iii. **New Members**:

Address	Service Port
10.1.20.11	80
10.1.20.12	80
10.1.20.13	80

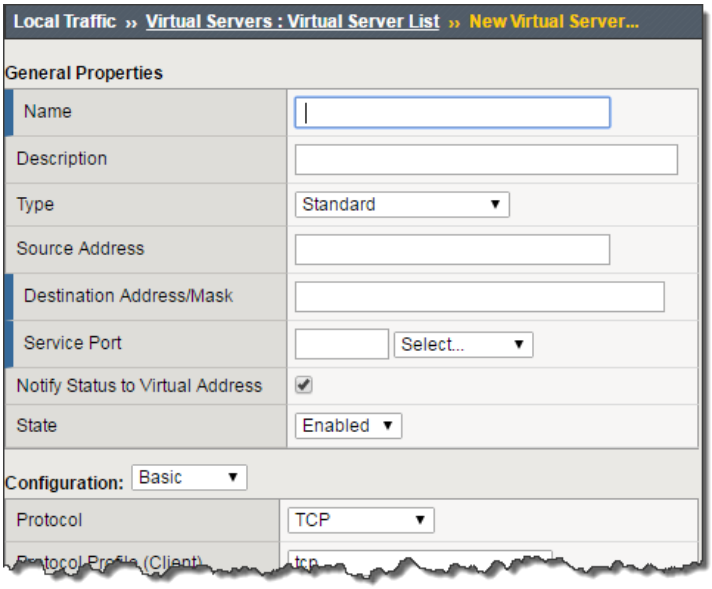
         1. As you enter each IP address and port combination hit **Add** button
   e. When you have complete your pool configuration, hit the **Finished** button



Creating Virtual Servers

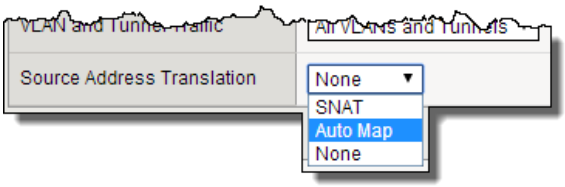
Now let's build our virtual server

3. Under **Local Traffic** select **Virtual Servers** then select **Create**.



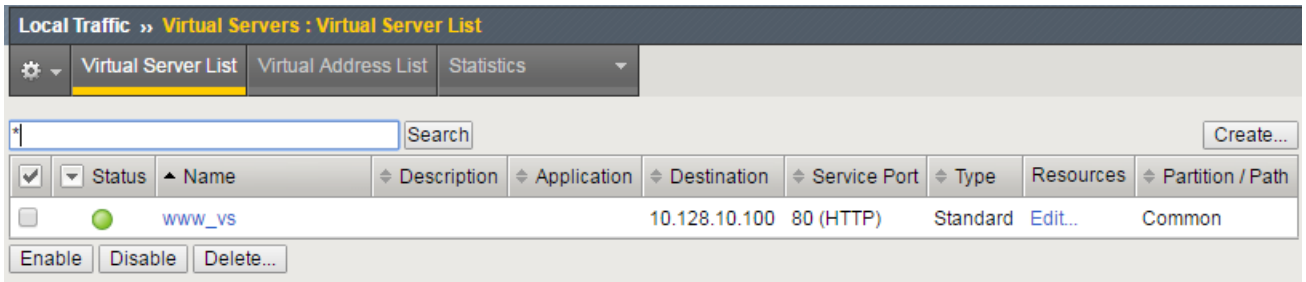
a. Under **General Properties**
   i. **Name:** www_vs
   ii. **Description**: <optional>
   iii. **Type:** Standard
   iv. **Source/Address:** <leave blank>
      1. **Note:** The default is 0.0.0.0/0, all source IP address are allowed
   v. **Destination Address/Mask:** 10.1.10.100
      1. NOTE: The default mask is /32
   vi. **Service Port**: 80 or HTTP

b. Under **Configurations**
   i. The web servers do not use the BIG-IP LTM as the default gateway. This means return traffic will route around the BIG-IP LTM and the TCP handshake will fail. To prevent this we can configure SNAT Automap on the Virtual Server. This will translate the client IP to the self IP of the egress VLAN and ensure the response returns to the BIG-IP.
   ii. **Source Address Translation**: Auto Map



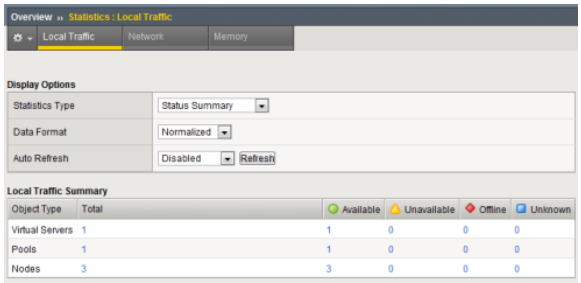
c. Under **Resources**
   i. **iRules**: none
   ii. **Policies**: none
   iii. **Default Pool**: From the drop down menu, select the pool (**www_pool**) which you created earlier
   iv. **Default Persistence Profile**: None
   v. **Fallback Persistence Profile**: None

4. When you have complete your pool configuration, hit the **Finished** button
5. You have now created a Virtual Server (Note: Items in blue are links)



6. Now let's see if our virtual server works!
   a. Open the browser to the Virtual Server you just created
   b. Refresh the browser screen several times (use "<ctrl>" F5)

   c. Go to your BIG-IP and view the statistics for the **www_vs** virtual server and the **www_pool** pool and its associated members
   d. Go to **Statistics > Module Statistics > Local Traffic**
      i. Choose **Virtual Servers** from drop down



   e. Go to **Local Traffic>Virtual Servers>Statistics**
   f. Go to **Local Traffic>Pools>Statistics**
      i. Did each pool member receive the same number of connections?
      ii. Did each pool member receive approximately the same number of bytes?
      iii. Note the Source and Destination address when you go to directly and through the virtual server

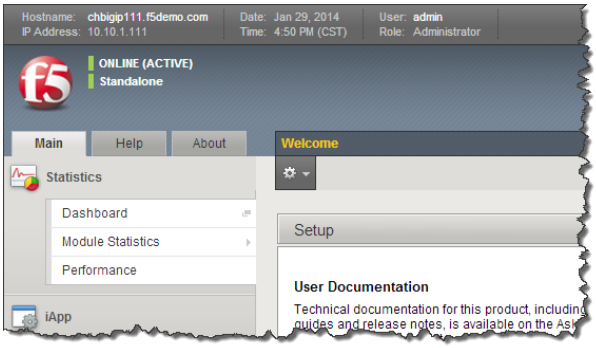
7. Let's archive our configuration in case we have to fall back later.
   a. Go to **System >> Archives** and select **Create**.
      i. Name your archive **lab2_the_basics_net_pool_vs**

ExtraCredit!

You can also review statistics via the CLI, simply SSH to the management IP of your BIG-IP. Refer to your Student Information page and Network Diagram for the IP address.

1. Check out the Linux CLI and TMSH
    a. **Username**: root **Password**: f5UDFrocks!
        i. Select VT100 as the terminal type
        ii. Review the information of the following commands:
        iii. bigtop –n
            1. Type **q** to quit.
    b. Take a look at the TMOS CLI, type "**tmsh**" to enter the Traffic Management Shell.
        i. (tmos)# **show ltm pool**
        ii. (tmos)# **show ltm pool detail**
            1. show statistics from all pools
        iii. (tmos)# **show ltm virtual**
        iv. (tmos)# **show ltm virtual detail**
            1. Show statistics of all virtual servers

2. Check out the Dashboard!
    a. Go to **Statistics>Dashboard**



3. Click the Big Red F5 ball. This will take you to the Welcome page. Here you can find links to:
    a. User Documentation, Running the Setup Utility, Support, Plug-ins, SNMP MIBs

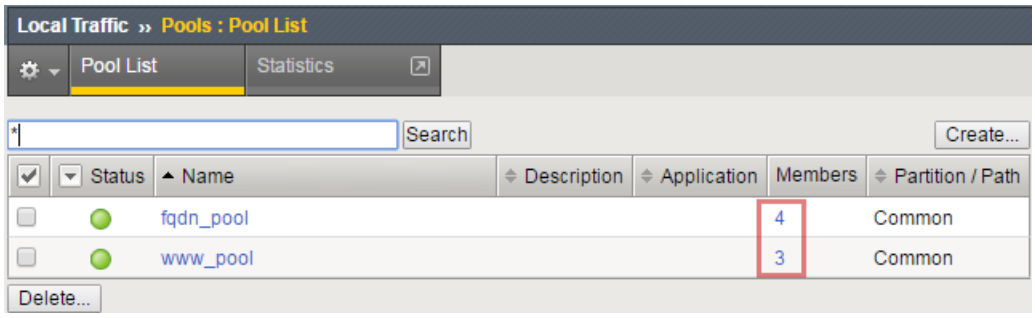
LAB 2: LOAD BALANCING, MONITORING AND PERSISTENCE

Objectives:

- Configure and review the Ratio load balancing method
- Build and test priority groups
- Build a content monitor that looks for a receive string and requires authentication
- Build and review simple (source IP) persistence and cookie persistence.

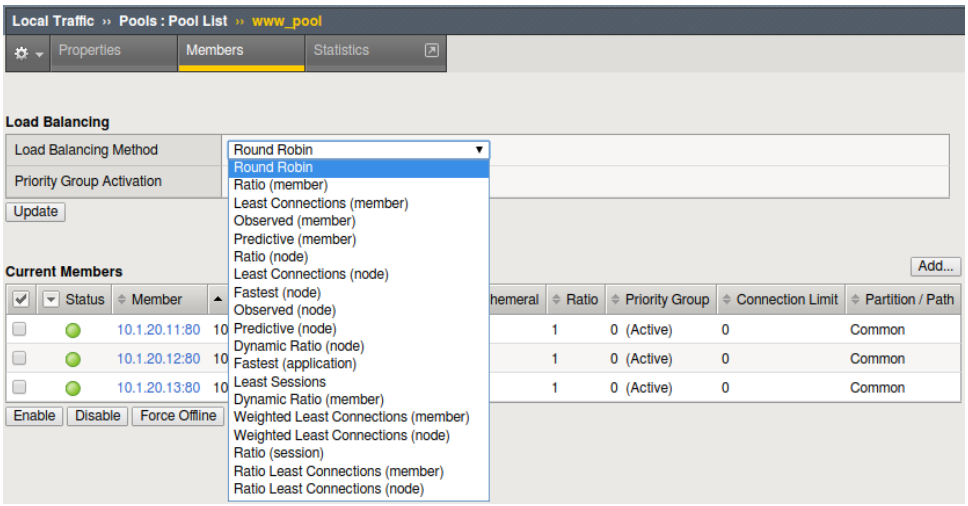
Ratio Load Balancing

4. Go to **Local Traffic>Pools** and select **www_pool** and then **Members** from the top bar or you could click on the **Members** link in the Pool List screen.



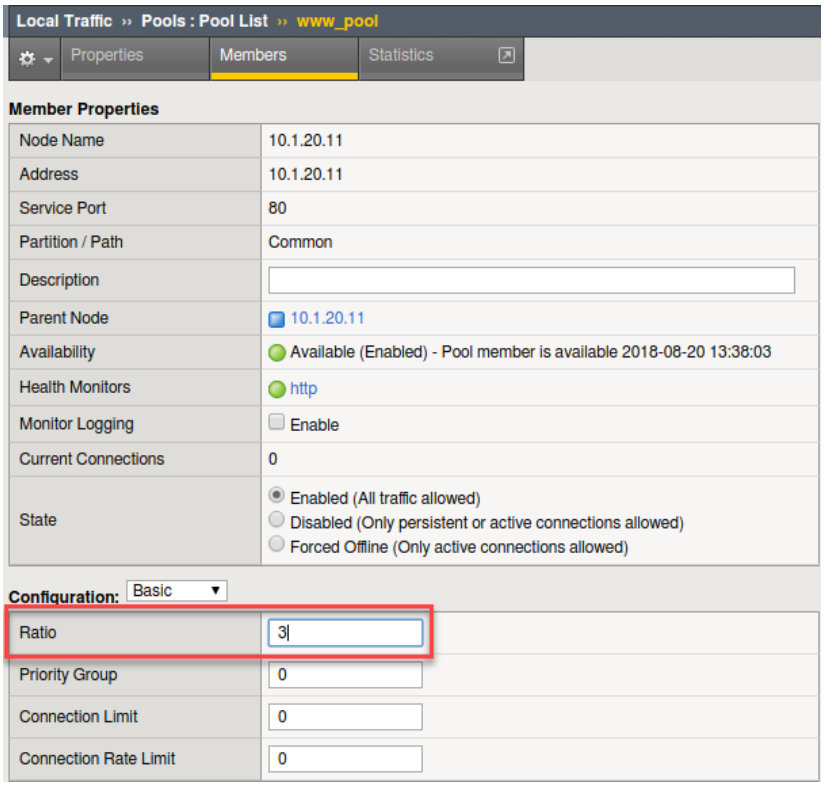
   a. Note: When we created the pool, we performed all our configuration on one page, but when we modify a pool the **Resource** information is under the **Members** tab

5. Under **Load Balancing** section
   a. Change the **Load Balancing Method** to **Ratio (Member)**
   b. As you look at the drop down menu, notice most load balancing methods have two options, (Node) or (Member). Remember the difference?



   c. Don't forget the **Update** button

d. Then under **Current Members**
   i. Select the first member in the pool **10.1.20.11:80**.
   ii. Under the **Configuration** section
      1. Change the **Ratio** of the member to 3



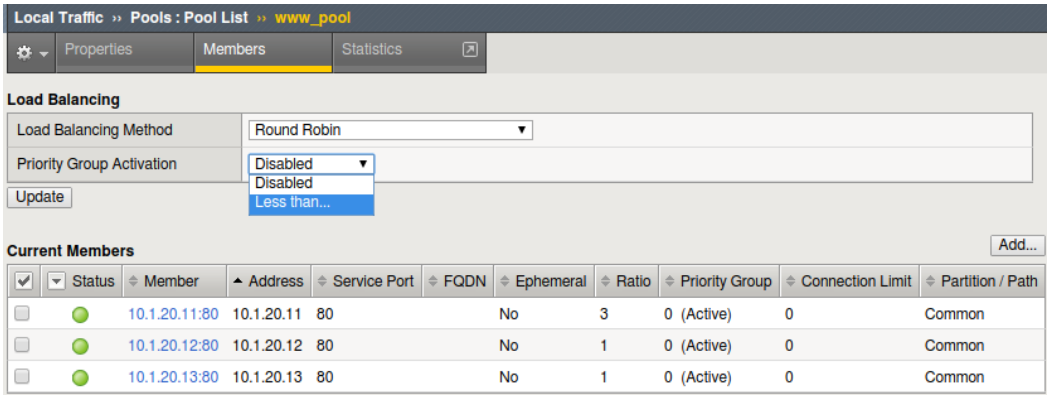
e. Select the **Update** button

6. Verification
   a. Check the pool statics by selecting **Statistics** in the top bar, if you are still in **Local Traffic> Pools** or by going to **Statistics>Module Statistics>Local Traffic** and selecting **Pool** from **Statistics Type**.
   b. Reset the statistics for your **www_pool** pool by checking the boxes next to the pool members and hitting the **Reset** button
      i. Browse to your **www_vs (10.1.10.100)** virtual server
      ii. Refresh the browser screen several times (use “<ctrl>” F5)
      iii. Select the **Refresh** button on the **Statistics** screen
      iv. How many Total connections has each member taken?
      v. Is the ratio of connections correct between the members?
   c. Now go back and put the pool back to Round Robin Load Balancing Method
      i. Reset the statistics
      ii. Refresh the virtual server page several times
      iii. Refresh the statistics
      iv. Does the ratio setting have any impact now?

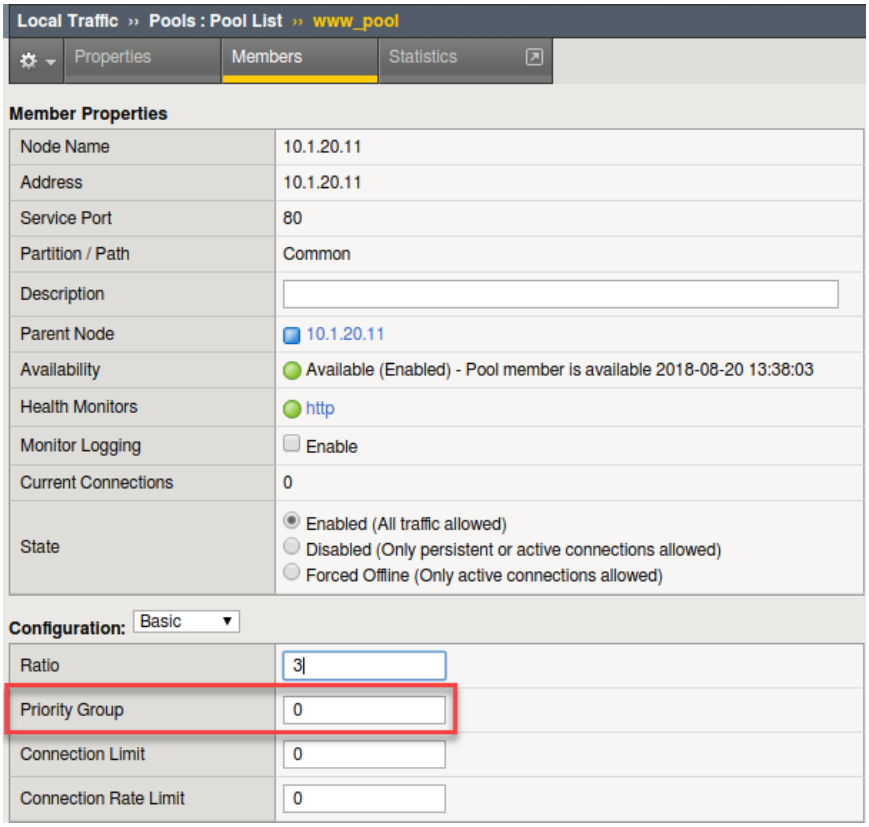
Priority Groups Lab

Let's look at priority groups. In this scenario we will treat the **.13** server as if it was is in a disaster recovery site that can be reached over a backhaul. To maintain at least two members in the pool for redundancy and load sharing, our customer would like to use it during maintenance periods or if one of the two other pool members fails.

1. Go to **Local Traffic>Pools>www_pool**
   a. Select the **Members** tab.
      i. Set the **Load Balancing Method** back to **Round Robin**
      ii. Set the **Priority Group Activation** to **Less than … 2** Available Members.



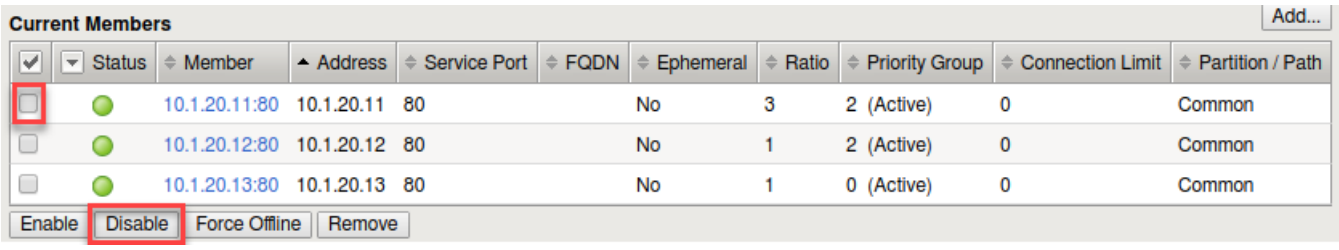
   b. Don't forget the **Update** button
   c. Select the pool members **10.1.20.11** and **10.1.20.12** and set their **Priority Group** to **2**.
      i. This will allow you to change the priority on that particular member.



2. Review your settings and let's see how load balancing reacts now.
    a. Select the **Statistics** tab.
    b. Reset the pool statistics.
    c. Browse to your virtual server and refresh several times.
    d. Refresh you statistics.
    e. Are all members taking connections?
    f. Which member isn't taking connections?

3. Let's simulate a maintenance window, or an outage, by disabling a pool member in the highest priority group. As this will drop the number of active members below 2, this should cause the low priority group to be activated.

4. Select the member in the Priority Group 2 and Disable that pool member.
    a. Select the **Disable** button



    b. The status indicator now goes to black, indicating the member has been disabled

5. Once again, select **Statistics**, reset the pool statistics, browse to the virtual server and see which pool members are taking hits now.

Once you are done testing re-enable your disabled pool member.

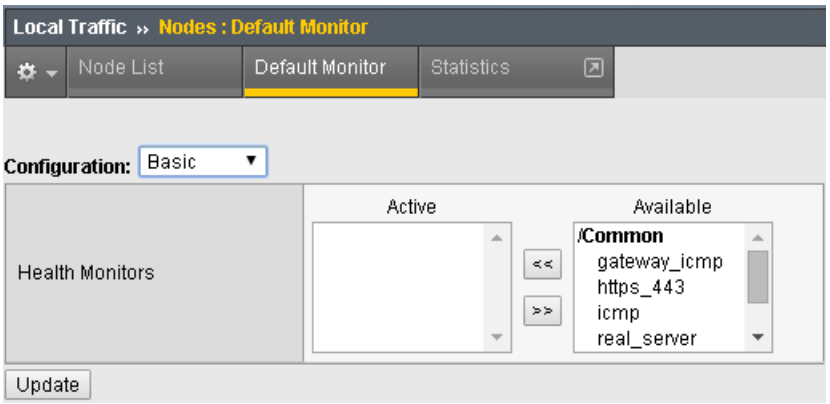
Monitor Labs

Objective:

- Build a default monitor for nodes
- Build a content monitor for your pool

Default Monitors

1. Go to **Local Traffic>Nodes**, note the Status.
   a. Notice there are Nodes in this table even though we never specifically configured them under the Node portion of the GUI. Each time a unique IP address is placed in a pool, by default, a corresponding node entry is added and assigned the default monitor (if any).
   b. Select the **Default Monitor** tab.



   c. Notice we have several options, for nodes you want a generic monitor, so we will choose **icmp**.
   d. Select **icmp** from the **Available** box and hit to place it in the **Active** box.
   e. Click on the **Update** button to finalize your changes.

2. Select **Node List** or **Statistics** from the top tab.
   a. What is the Status of the Nodes?

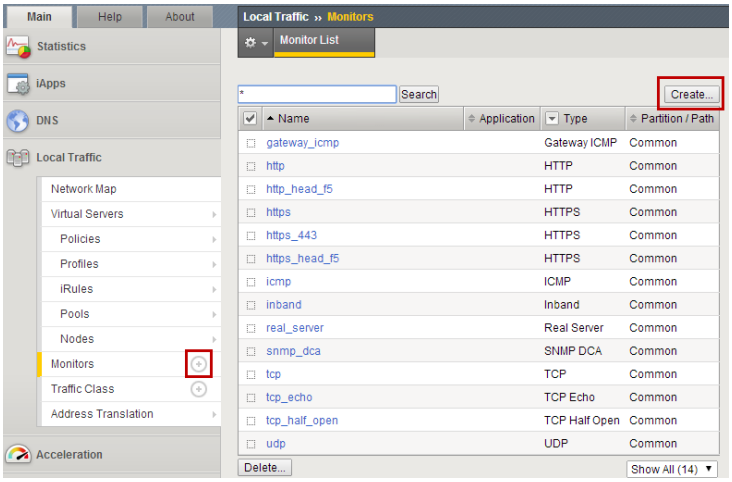
3. Select **Statistics>Module Statistics>Local Traffic**
   a. What is the Status of your Nodes, Pool and Virtual Server?

Content Monitors

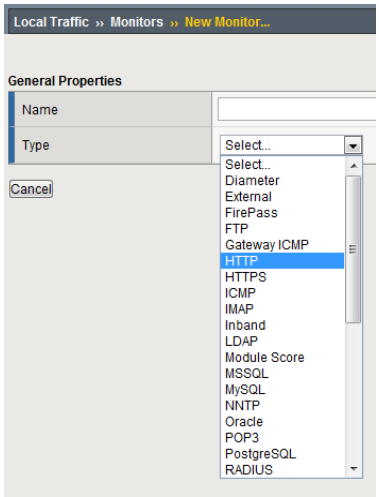
The default monitor simply tells us the IP address is accessible, but we really don't know the status of the particular application the node supports. We are now going to create a monitor to specifically test the application we are interested in. We are going to check our web site and its basic authentication capabilities.

4. Browse to **http://10.1.10.100** and on the web page select the **Basic Authentication** link under **Authentication Examples**.
   a. User: **user.1**
   b. Password: **password**
   c. You could use text from this page or text within the source code to test for availability. You could also use HTTP statuses or header information. You will be looking for the HTTP status "**200 OK**" as our receive string to determine availability.
   d. Note the URI is **/basic**. You will need this for your monitor.

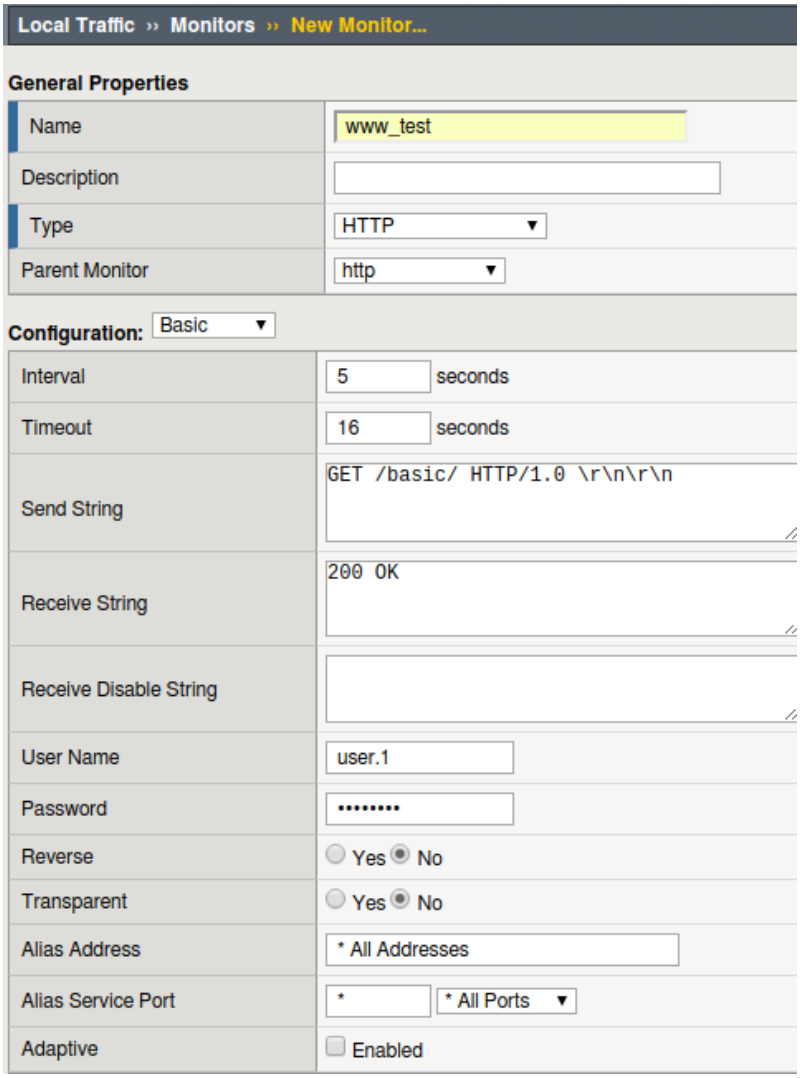
5. Select **Local Traffic>Monitor** on the side-bar and select the plus (**+**) sign or the **Create**



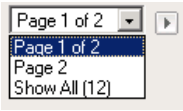
   a. Now we can create a monitor to check the content of our web page to ensure things are running properly.
      i. **Name**: www_test
      ii. **Type**: HTTP



b. Once you have selected your parent (Type) monitor, you can access the **Configuration** section
   i. **Send String**: Enter the command to retrieve the page you want "**GET /basic/ HTTP/1.0 \r\n\r\n**" (no quotes)
   ii. In the Receive String box put "**200 OK**" (no quotes)
      1. **NOTE**: The receive string is not case sensitive.
   iii. Enter **user.1/password** for the **Username** and **Password**

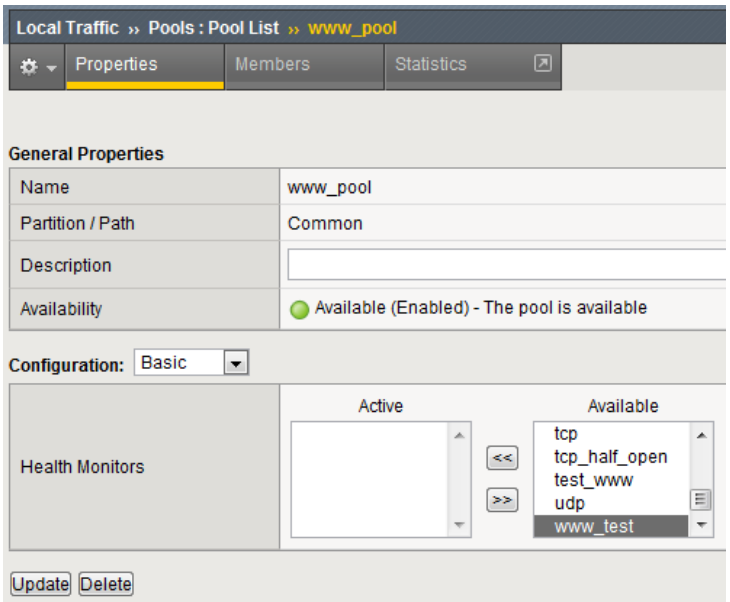


c. Click **Finish** and you will be taken back to **Local Traffic>Monitors**

6. Where is your new Monitor?
   a. **Hint:** Check the lower right hand corner of the Monitors list, 

here you can go to the next page or view all Monitors
   b. You can change the number of records displayed per page in **System>Preferences**

7. Go to **Local Traffic>Pools>www_pool** and choose **Properties** from the top bar.
    a. Remove the **http** monitor from the Active box.
    b. Select the **www_test** monitor from the Available monitor's window in the **Configuration** section and move it to the Active window.



8. Hit **Update** to apply the change.
    a. Select **Statistics** from the tabs.
    b. What is the status of the pool and its members?

9. Go to **Local Traffic>Virtual Servers**, what is the status of your virtual server?
    a. Browse to your **www_vs** virtual server. Which members are taking traffic?
    b. Just for fun reverse the monitor. Now when **200 OK** is returned it indicates the server is not responding successfully. You can see where this would be useful if you were looking for a 404 (bad page) response.

Persistence Labs

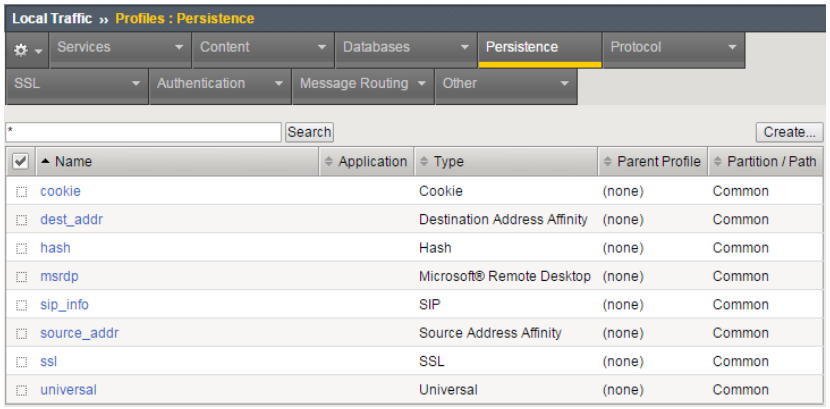
In this lab we will configure a couple types of persistence and view their behavior. For persistence, profiles will have to be created and attached to our virtual server.

Lab Requirements:

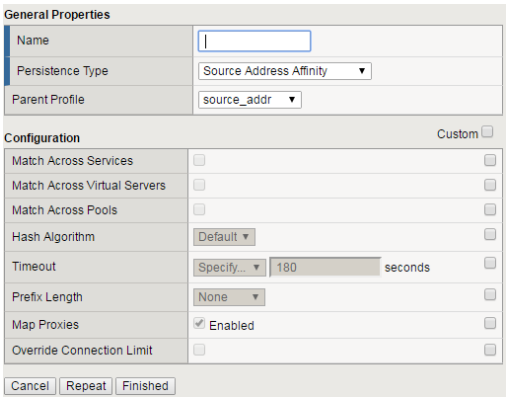
- Prior to beginning the lab verify your **www_pool** has been set to the following parameters:
  - **Load Balancing Method**: Round Robin
  - **Priority Group Activation**: Disable
    - The members **Ratio** and **Priority Group** mean nothing since we aren't using Ratio load balancing and Priority Groups are disabled.
  - Hit **Update**
  - Hit your virtual server several times, you should see all 3 servers respond.

Simple (Source Address) Persistence

1. Go to **Local Traffic>Profiles** and select the **Persistence** tab.
   a. From the **Persistence Profiles** screen select the **Create** button.



   b. At the **New Persistence Profile** screen enter:
      i. **Name**: my-src-persist
      ii. **Persistence Type**: Source Address Affinity



   c. This will add the **Configuration** section to the **General Properties** section.
      i. Note the parent profile.

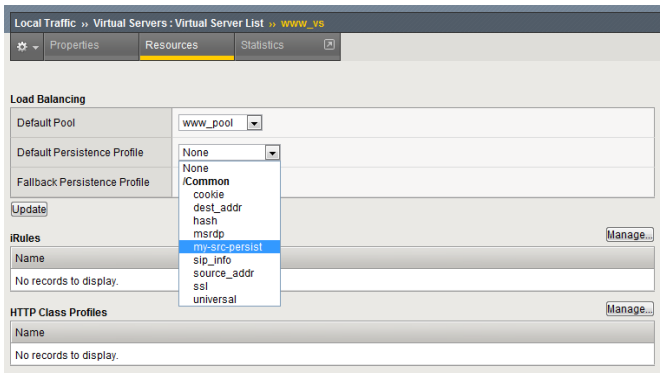
d. In the **Configuration** section, set the
   i. **Timeout**: 60 seconds
   ii. **Prefix Length**: None
      1. This is the default, and is a /32 prefix (255.255.255.255 mask).
      2. Each new IP address will create a new persistence record.
   iii. **Hint**: You can't change the settings until you have checked the Custom box.
      1. Hey, I didn't write the GUI, but actually this is very useful in knowing which configuration items were modified from the default.
   iv. Click the Finished button.

e. You have just created your first custom Profile.
   i. Note the check box for your new custom profile isn't grayed out and can be selected to allow you to delete the profile if desired.

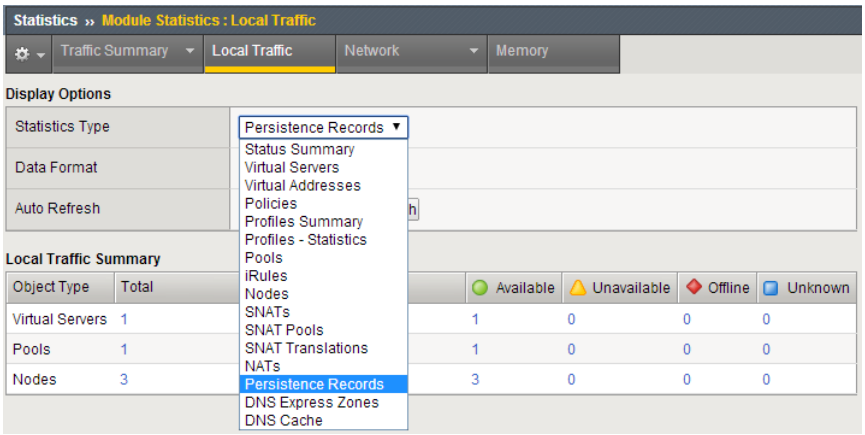
2. Now let's attach our new profile to the virtual server.
   a. Go to **Local Traffic>Virtual Server** and ….
      i. Select **www_vs** and the **Resources** tab or ….
      ii. Take the shortcut directly to the **Resources** of the virtual server. (Can you find it?)

Note: When we created the Virtual Server everything was on a single page, we find when we return to modify the Virtual Server the Properties and Resources are on different pages.

   b. Set the **Default Persistence Profile** to **my-src-persist**.

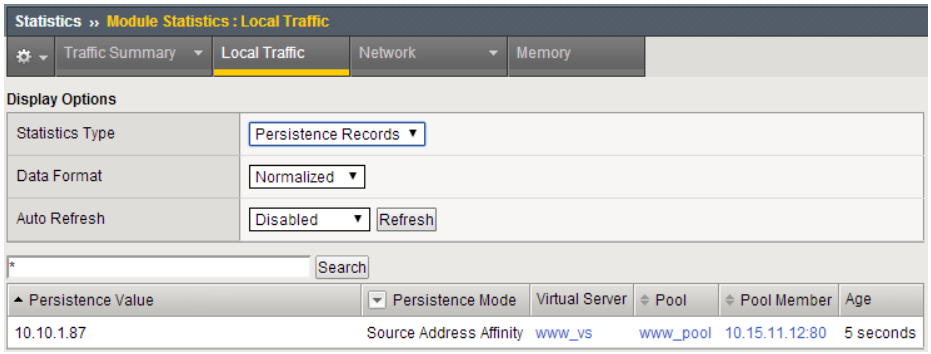


   c. Don't forget to **Update** before leaving the page. *(Be careful, someday I will quit telling you that.)*
   d. Testing Source Address Affinity
      i. At this point you may want to open a second browser window to the management GUI.
      ii. For one management window go to **Statistics>Module Statistic>Local Traffic**
      iii. Select **Persistence Records** for the **Statistics Type** menu



3. At this point you will see that Persistence Records statistics display has been disabled in version 12.1. A TMSH database command is required to activate it.
    a. SSH to you BIG-IP at 10.1.1.245. Username: **root** Password: **f5UDFrocks!**
    b. At the prompt enter: **tmsh**
    c. At the TMSH prompt enter the command in the **Persistence Value** GUI.
        i. **modify sys db ui.statistics.modulestatistics.localtraffic.persistencerecords value true**
            1. Tab completion will make this a little easier

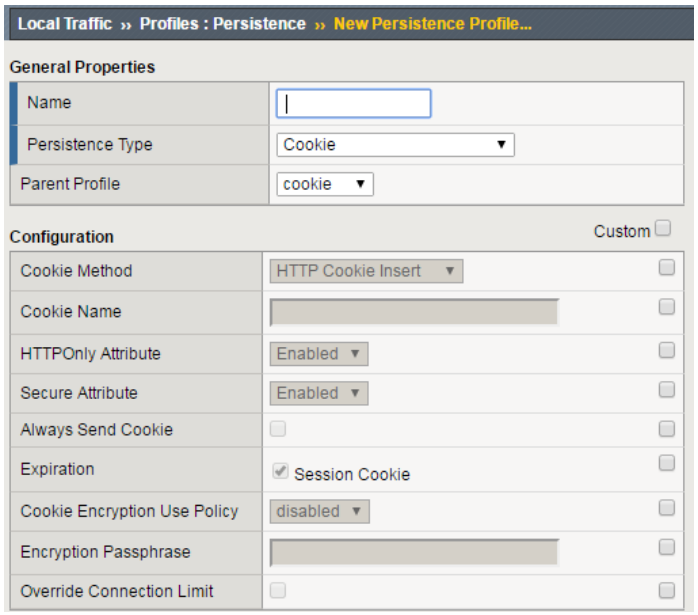
4. Now, in this window you can watch you persistence records. You may want to set **Auto Refresh** to 20 seconds.



5. In your other management GUI window go to **www_pool** and clear the member statistics.
    a. Open a browser session to your virtual server and refresh several times.
    b. How many members are taking traffic?
    c. Check your Persists Records window, are the any persistence records?
        i. If you are not Auto Refreshing, don't forget to hit Refresh
    d. Refresh your web page prior to the Age column reaching 60. What happens?

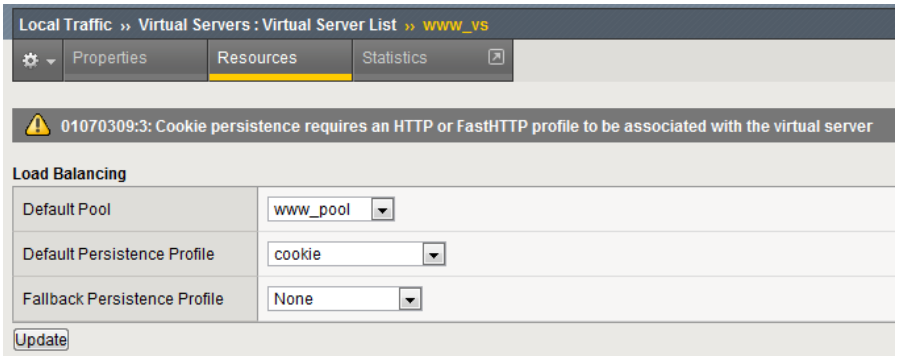
Cookie Persistence (Cookie Insert)

1. Go to **Local Traffic>Profiles>Persistence** tab and hit **Create**
    a. Let's name our profile **my_cookie_insert** (original isn't it)
    b. Our **Persistence Type** will be **Cookie**
    c. This brings us to the **Configuration** section.



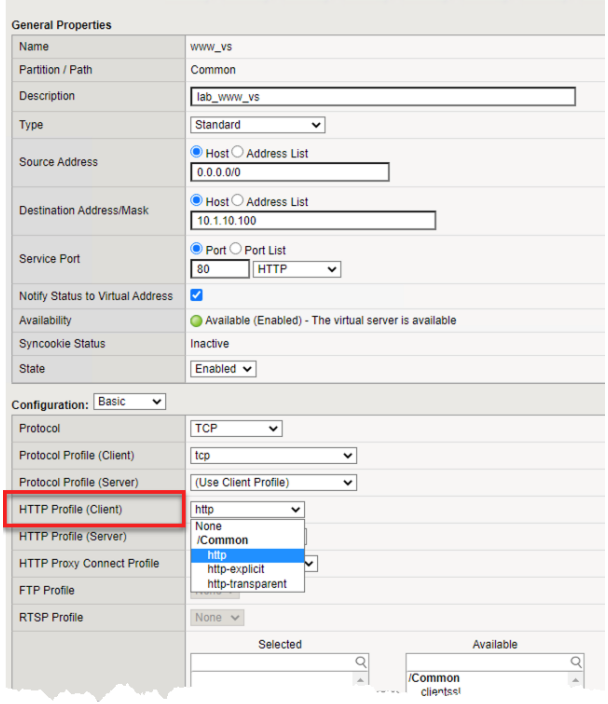
2. As you can see the default **Cookie Method** is **HTTP Cookie Insert**, so we won't have to modify the Cookie Method
    a. The BIG-IP will also create a cookie name for you using a combination of "**BIGipServer**" and the pool name the virtual server service. We will take this default also.
    b. We will use a session cookie. Which means the cookie is deleted when the browser is closed.
    c. Select **Finished**
    d. Now attach your cookie persistence profile to your virtual server's **Default Persistence Profile**

    Go to **Local Traffic>Virtual Server>www_vs>Resources** tab
    e. Set the **Default Persistence Profile** to **my_cookie_insert**
    f. Hit **Update**
    g. Whoa! Did you just get this error message?



    h. Remember what we said earlier about some Profiles requiring prerequisite Profiles? Since we are looking in the HTTP header for the cookie the prerequisite for the Cookie Profile is the HTTP profile.

3. We will have to go to the virtual server to add the HTTP profile, prior to adding the Cookie Persistence profile.
    a. Select the **Properties** tab on your virtual server
    b. Go to **HTTP Profile** in the **Configuration** section and select the default HTTP (**http**) profile.



    c. Hit the **Update** button
    d. Now we can go back to the **Resource** tab and add our cookie persistence profile.

4. Testing cookie persistence.
    a. If you wish you can watch the member statistics to validate your persistence.
    b. Open a new browser session to your virtual server and refresh several times.
    c. Does the page ever change?
    d. Did you hit a different server?
    e. Refresh several times. Are you hitting the same server?
        i. On the web page under **HTTP Request and Response Information** click the **Request and Response Header** link.

Archive your work in the file: **lab3_lb_monitor_and_persist**

LAB 3: ACCELERATING APPLICATIONS LAB

Objectives:

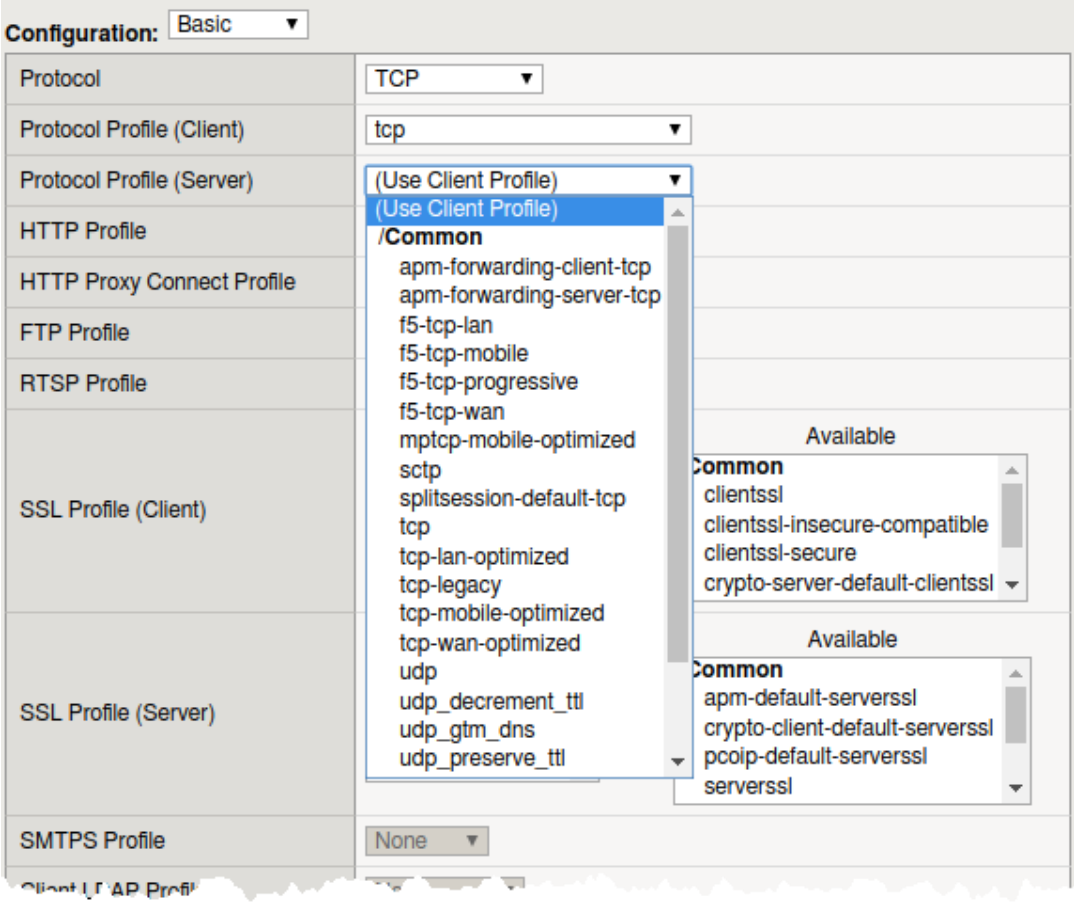
- Assign client-side and server-side profiles
- Set up caching for your web site
- Set up compression for your web site

Lab Prerequisites:

- Prior to starting this lab remove the cookie persistence profile from you virtual server.

TCP Express

1. Set clientside and serverside TCP profiles on your virtual server properties.
    a. If you chose to use the **Advaced** menu you will see a whole array of new options. There are **Basic** and **Advanced** drop downs on many of the GUI menus. You can always see **Advanced** menus by changing the preferences in **System>Preferences.**
    b. From the dropdown menus place the **tcp-wan-optimized** profile on the client-side and the **tcp-lan-optimized** profile on the server-side.



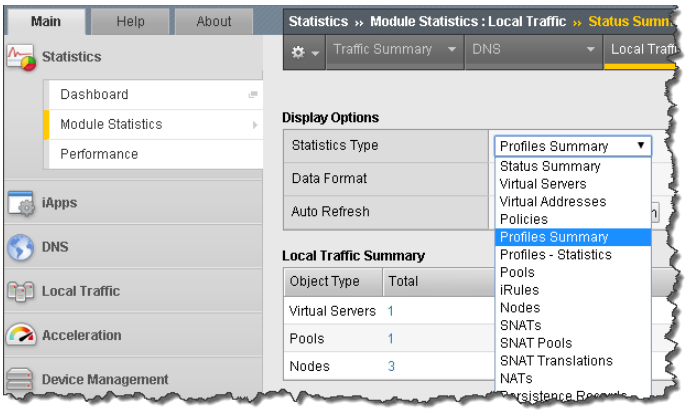
HTTP Optimization - RamCache Lab

1. Go to your virtual server and refresh server times. Note the Source Node for the pictures of the BIG-IPs. They change depending on where the connection is coming from. The Source Node information is part of the picture.

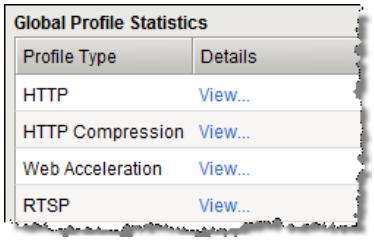
2. Go to **Local Traffic>Profiles>Services>Web Acceleration** or **Acceleration>Profiles>Web Acceleration**
   a. Create a new profile named **www-opt-caching** using **optimized-caching** as the Parent Profile.
   b. Take all the defaults, no other changes are required.

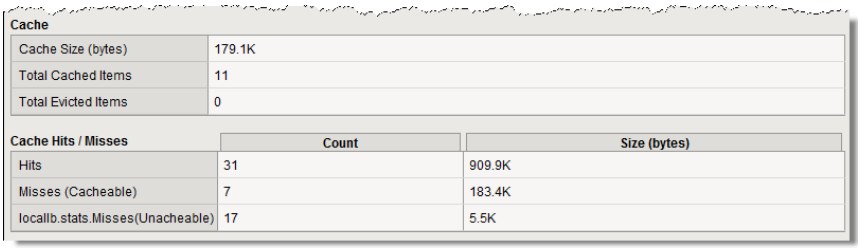
3. Open up your **www_vs** virtual server.
   a. At the **HTTP Profile** drop down menu make sure **http** is selected.
   b. Under **Acceleration** at **Web Acceleration Profile** select your new caching profile; **www-opt-caching**
   c. Clear the statistics on your pool and the refresh the main web page several times.
      ii. The pictures do not change. Why do you think that is?
      iii. Go to your pool. Are all pool members taking connections?

4. Now go to **Statistics>Module Statistics>Local Traffic** on the sidebar, from the **Statistics Type** drop down menu select **Profiles Summary**



5. Select the View link next to the **Web Acceleration** profile type





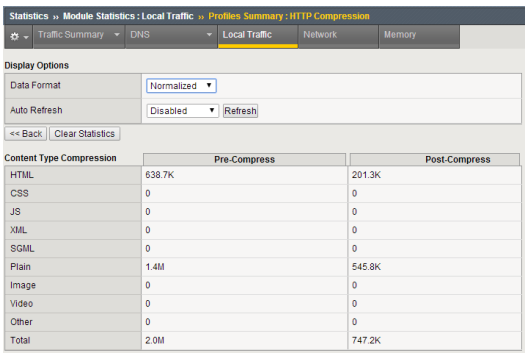
6. You can get more detailed information on ramcache entries at the CLI level
    a. Log onto the CLI of your BIG-IP via SSH using the root account (user: **root** password: **f5UDFrocks!**).
    b. At the CLI go into **tmsh** at the **(tmos)#** prompt
    c. At the shell prompt enter **show ltm profile ramcache www-opt-caching**

HTTP Optimization - HTTP Compression Lab

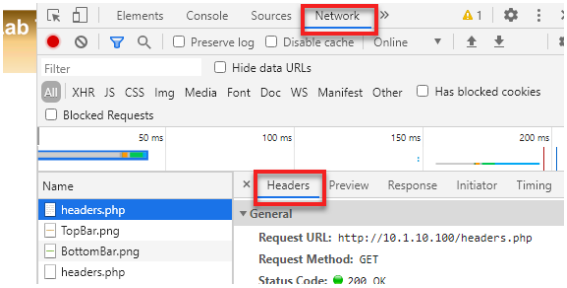
1. Go to **Local Traffic>Profiles>Service>HTTP Compression** or **Acceleration>Profiles>Web Acceleration**
    a. Create a new profile, **www-compress** using the **wan-optimized-compression** default profile.

2. Open up your **www_vs** virtual server.
    a. At the **HTTP Profile** drop down menu make sure **http** is selected.
    b. At the **Web Acceleration** drop down menu select **None**
        iv. *For purpose of this lab we don't want caching interfering with our response headers*.
    c. At the **HTTP Compression** drop down menu select the HTTP compression profile you just created

2. Now hit your virtual server and on the web page under **Content Examples on This Host** select the **HTTP Compress Example** and **Plaintext Compress Example** link.
    a. Now off to the statistics on the sidebar, under the **Local Traffic** drop down menu select **Profiles Summary**
    b. Select the **View** link next to the **HTTP Compression** profile type



    c. On the web page under, **HTTP Request and Response Information** select the **Request and Response Headers** link. Notice you no longer see the **Accept-Encoding** header in the **Request Headers Received at the Server** section.
        i. Alternately you can right click in the Chrome window and select **Inspect**
            1. Select **Network** from the top bar in the Inspect window.
            2. Refresh the page and select the **.php** pager and **Headers** on the bar to the right.



    d. You can also browse directly to one to the pool members, to help you find what has changed.

Archive your work in a file called: **lb4_acceleration**

LAB 4: SSL OFFLOAD AND SECURITY

In this Lab we will configure client side SSL processing on the BIG-IP.

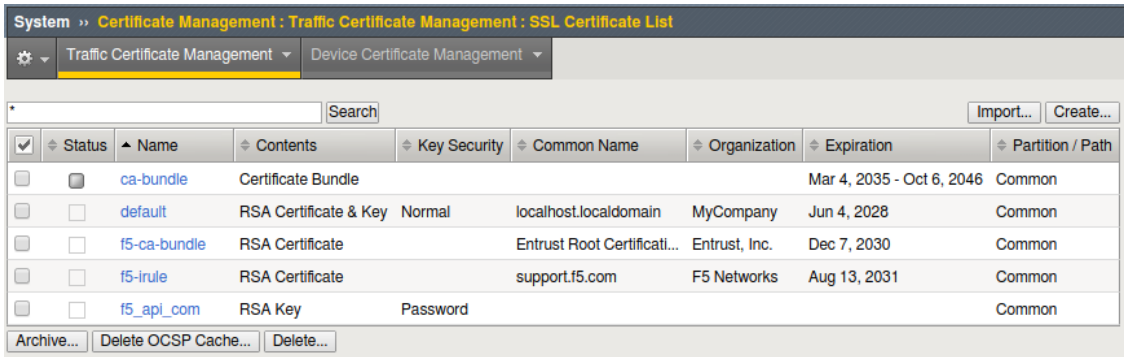
Objective:

- Create a self-signed certificate
- Create a client SSL profile
- Modify your HTTP virtual server to use HTTPS
- Add addition security to your HTTPS web server using the HTTP profile

We will create a self-signed certificate and key and a SSL client profile to attach to our virtual server.

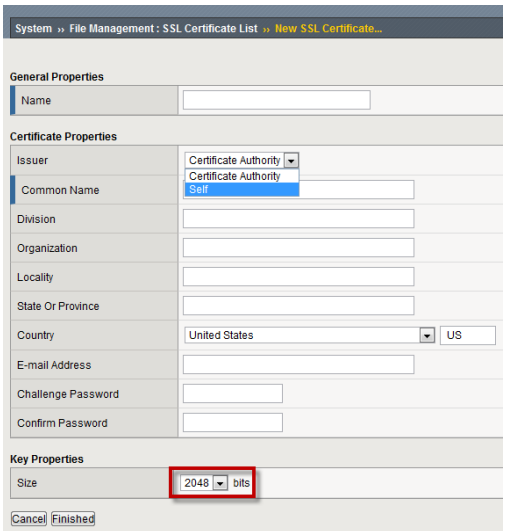
Creating a Self-signed certificate and key

1. Go to **System > Certificate Management > Traffic Certificate Management > SSL Certificates List** and select **Create**



This will take you to **Local Traffic >> SSL Certificates >> New SSL Certificate…**

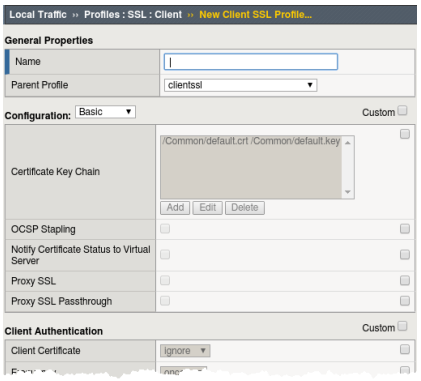
a. NOTE: The default key size is **2048**, you can save SSL resources on the **server-side** by lowering this key size.



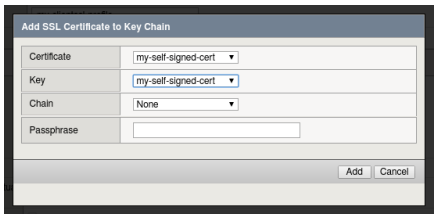
b. Enter:
   i. **Name**: my-selfsigned-cert
   ii. **Issuer**: Self
   iii. **Common Name**: www.f5demo.com
   iv. Fill out the rest any way you would like.

Creating SSL Client Profile

2. Go to **Local Traffic>Profiles>SSL>Client** menu and select **Create**.



   a. Under **General Properties**
      i. **Name**: my_clientssl_profile
   b. Under **Configuration** in the **Certificate Key Chain** section, select the **Custom** box and hit **Add**.
      i. In the **Add SSL Certificate to Key Chain** pop-up select:
         1. **Certificate**: my-selfsigned-cert
         2. **Key**: my-selfsigned-cert
      ii. Select **Add**



   c. Hit **Finished.**

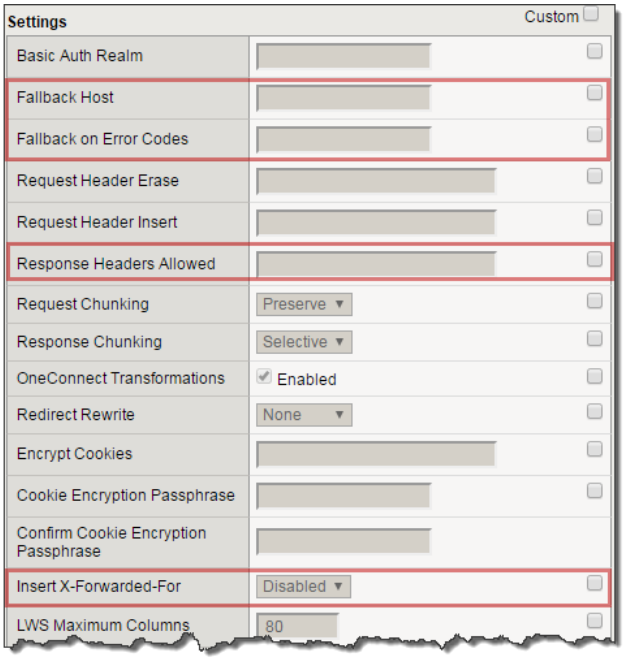
Building our New Secure Virtual Server

1. Go to **Local Traffic>Virtual Servers** and hit the Create button or hit the "+" next to Virtual Servers
   a. **Name**: secure_vs
   b. **Destination Address/Mask**: 10.1.10.105
   c. **Port**: 443 or HTTPS
   d. **SSL Profile (Client)**: my_clientssl_profile (the profile you just created)
   e. **Source Address Translation**: Auto Map (remember why we need this?)
   f. **Default Pool**: www_pool
   g. Default all other settings. (Notice you did not require an HTTP profile)
   h. **Finish**

2. Testing our secure server. Go to you **secure_vs** at **https://10.1.10.105**
   a. If you want to watch member traffic, go to the **www_pool** and reset the statistics.
   b. Browse to your secure virtual server
   c. What port did you pool members see traffic on?

Securing web applications with the HTTP profile

1. Let's begin by creating a custom HTTP profile.
    a. Go to **Local Traffic> Profiles>Services**, select HTTP create a new profile
    b. Under **General Properties**
        i. **Name**: secure-my-website
    c. Under **Settings**:
        i. Set the **Fallback Host**: http://10.1.1.252 *(this will take you an internal site)*
        ii. **Fallback on Error Codes**: 404 (fallback site if a 404 error is received)
        iii. **Response Headers Allowed**: Content-Type Set-Cookie Location
        iv. **Insert XForwarded For**: Enabled (because we talked about it earlier)



    d. Attach your new HTTP Profile to your secure (HTTPS) virtual server

2. Browse to your secure virtual server.
    a. Do web pages appear normal?
    b. Now browse to a bad page.
    c. For example, https://10.1.10.105 /badpage
        i. What is the result?
    d. Go to the **Request and Response Headers** page, you should see a sanitized server response at the bottom of the web page and the original client IP address.
    e. You can compare the headers by accessing your HTTP virtual server at http://10.1.10.100.
    f. While you are looking at the headers, check for the **X-Forwarded-For** header received by the server

    NOTE: Even though the data is encrypted between your browser and the virtual server, the LTM can still modify the data (i.e. resource cloaking) because the data is unencrypted and decompressed within TMOS.

Archive your work in a file called: **lab5_security**

LAB 5: BIG-IP POLICIES AND IRULES

In your customers environment the web servers retrieve images from a different set of servers. In the lab you will write and iRule and create a BIG-IP policies so you can compare and contrast the to methods. iRules are more flexible and customizable, while BIG-IP policies are easier to use, require no coding skills and are a little more efficient when performing the same task.

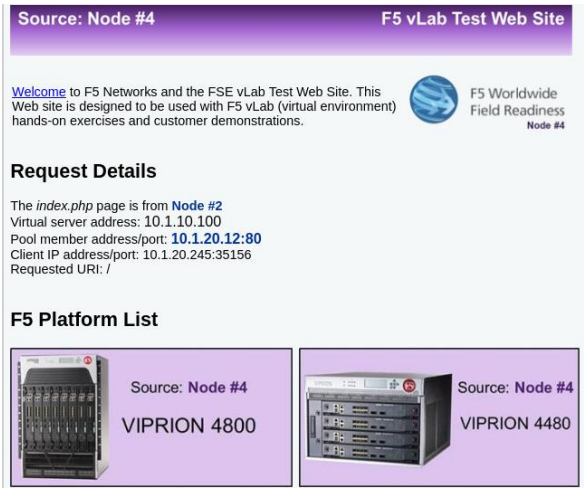
Write an iRule to retrieve images when an HTTP request is received

When HTTP request is received, look at the HTTP URI. If the URI ends with **jpg** or **png** send the request to an alternate pool of image servers.

1. **Create** a new pool named **image_pool**, use the **http** monitor for status and add one member **10.1.20.14:80**.
2. Go to **Local Traffic > iRules > iRules List** and select the create button.
   a. **Names:** retrieve_images
   b. **Definition:**

```
when HTTP_REQUEST {
# If the content is a jpeg or portable graphic (png) go to the image pool
if { ([HTTP::uri] ends_with "jpg") or ([HTTP::uri] ends_with "png") } {
   pool image_pool
  }
}
```

   c. Note the highlighted content, click on HTTP_REQUEST and HTTP::uri to get information on the event and command.
3. Go to **Local Traffic > Virtual Servers** and open the **secure_vs** virtual server. Go to the **Resources** section.
   a. Under **iRules** select the **Manage** button and put the **retrieve_images** iRule into the **Enabled** box and add the iRule to the virtual server.
      i. What other profile did this iRule require to work?
4. Test your policy by going to **https://10.1.10.105**, you will want to use an incognito/private browsing window to avoid cached content.
   a. All you image content should come from Node 4 (10.1.20.14).



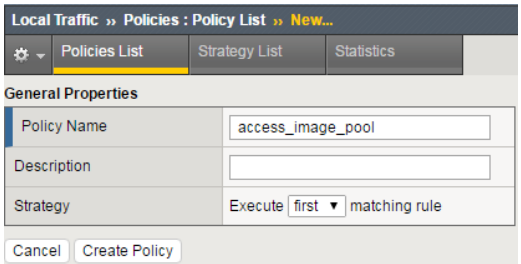
   b. Where is non-image request go?

Use a BIG-IP Policy to retrieve images from a different pool

In this task you are going to the same thing as above, except you will use a BIG-IP policy.

1. First you create your policy container and set your match strategy. Try to do this using the instructions, but a screen shot of the policy is available in the **Appendix** at the end of the lab guide if you would like it.

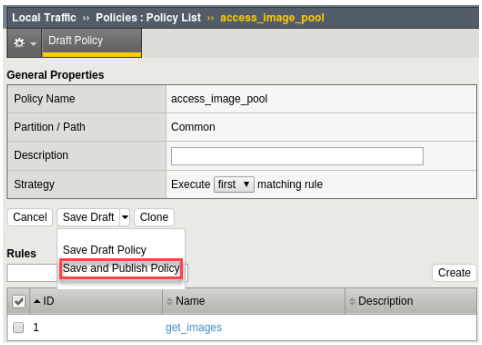
2. Go to **Local Traffic ›› Policies : Policy List** and select **Create**
   a. **Policy_Name:** access_image_pool
   b. **Strategy:** Execute **first** matching rule.
   c. **Create Policy**



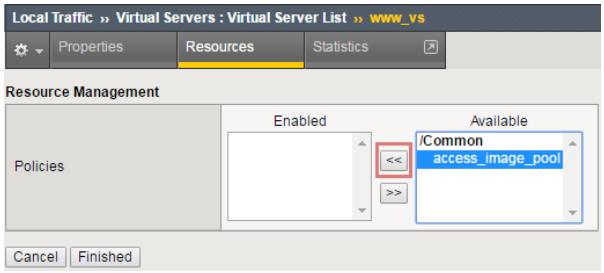
3. Now you can create/view policy rules. Select **Create**.
   a. **Name:** get_images
   b. In the box under **Match all the following conditions:** select the + to the right of **All traffic**
      i. Use the top drop down menu to select **HTTP URI**, on the next line of dropdown boxes select:
         1. **extension ends_with any of <Add jpg & png>** at **request** time
   c. Under **Do the following when the traffic is matched:** build the following operation.
      i. **Forward Traffic** to **pool Common/image_pool** at **request** time.

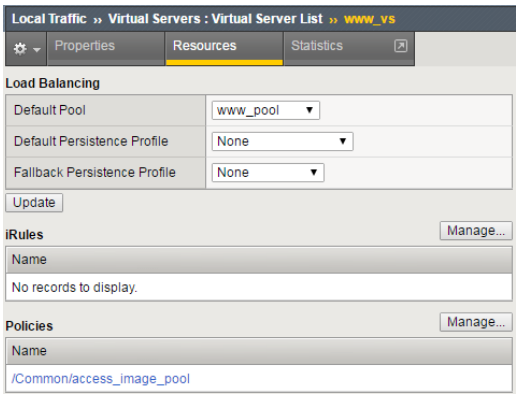
   d. **Save**

4. The policy is saved in **Draft** form and is not available/update until **Published**. To publish the policy:
   a. Select the **Save Draft Policy** drop-down menu and select **Save and Publish Policy**.

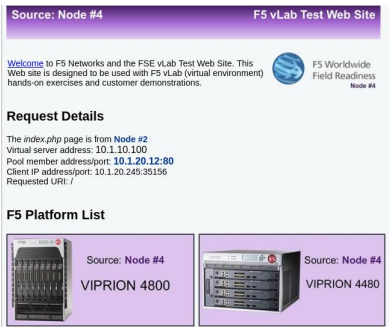


5. Go to the **Resources** section of your **secure_vs** virtual server and select **Managed** over the **Policies** box.
    a. Remove the **retrieves_images** irule from the virtual server.
    b. Move **access_image_pool** for the **Available** box to the **Enabled** box.





6. Now test your change by browsing to **http://10.1.10.105.**
    c. If your policy is working is working correctly all the images under **F5 Platform List** should be from **NODE #4**.
    d. Other images are PNG images and have a different extension.



LAB 6: SUPPORT AND TROUBLESHOOTING

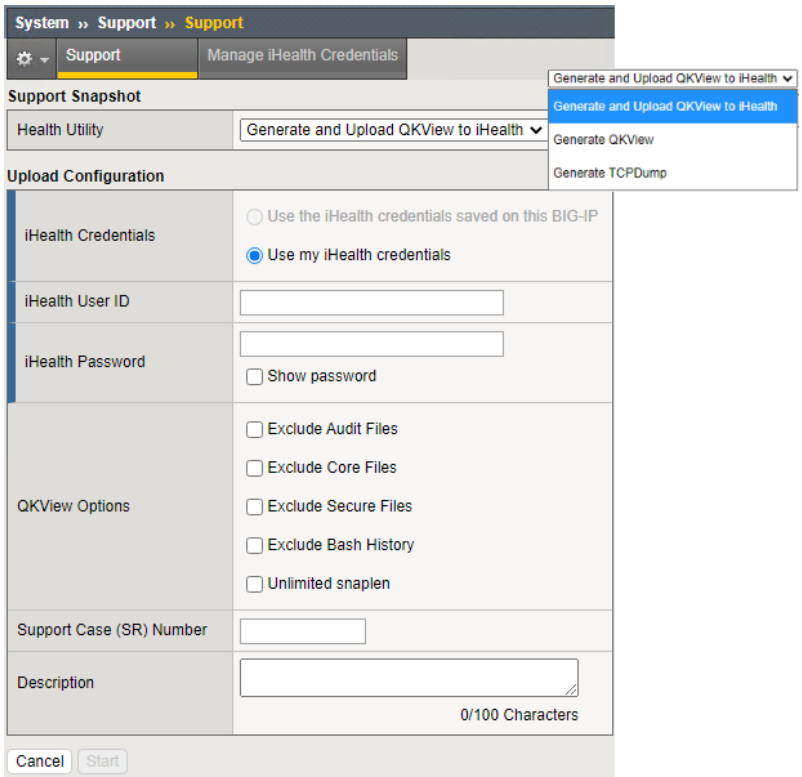
In this lab you will review your BIG-IP using iHealth and perform some basic troubleshooting commands

Objective:

- Get a QKView and upload it to http://ihealth.f5.com and review the results.
- Perform a TCPDump to watch traffic flow.
- Obtain web page information via Curl.

Archive the current configuration and perform a health check using a QKview

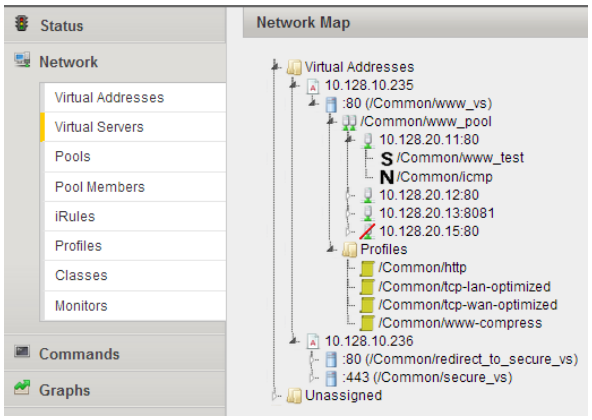
1. Obtain a **QKView**. Go to **System > Support**
    a. Here under **System>Support>Manage iHealth Credentials** you can enter you iHealth (F5) credentials
    c. From **System>Support** select the **New Support Snapshot** button to create a QKView
        a. From here you can create and upload a qkview, just create a qkview or create a TCPDump



2. Import the QKView into iHealth if you did not automatically upload the QKview to iHealth.
    d. Go to **http://ihealth.f5.com**. **If you don't have an account, now is the time to create one and then skip to the next section "Troubleshoot using TCPDump or Curl" because it will take time for your account set up.**
    e. Select **Upload to iHealth** button and upload the QKView file you download from your BIG-IP
    f. Once the file is uploaded you can click on the hostname to view you the heuristics.
        i. Note the Diagnostics. Go to **Diagnostics > Critical** on the side-bar.
            1. Example: **The configuration contains user accounts with insecure passwords** is because we are using default passwords.

ii. Select **Network > Virtual Servers**, then click on the small white triangles to expand the view or go to **Pools**, then **Pool Members** to continue to expand the view.

1. This is a little more detailed than **Local Traffic > Network Map**



g. Want to know interesting CLI commands, go to **Commands > Standard** and expand **tmsh** then **LTM** and click on **show /ltm virtual** toward the bottom.

h. Under **Files > config** you can view the **bigip.conf** file and see the command lines you used for you build.

iii. All the **log** files are here too

i. Feel free to just poke around.

Troubleshoot using TCPDump or Curl.

1. Go to your **www_vs (10.1.10.100)** virtual server and set **Source Address Translation** to **None**.
   a. Now browse the web site. You will be able to access it even though the status of the virtual is available.
      i. Because BIG-IP is not the server's default gateway of the servers their response goes around the BIG-IP.
   b. The web administrator tells you everything is fine as far as he can see and thinks the issue is with the BIG-IP, because they ALWAYS think the issue is with the BIG-IP.
   c. You begin by debugging the client connections to the web servers through the BIG-IP using TCPDump.

2. SSH to the management port of your BIG-IP. Remember the BIG-IP is a full proxy. You will need two dumps and therefore two SSH windows for the client-side connection and the server-side connection.
   a. First let's see what if we are hitting the virtual server. At the Linux CLI prompt:
      i. tcpdump –i <client vlan name> host –X –s128 10.1.10.100 and port 80
         1. This is a little overkill, but a good example of syntax. We will only look at traffic headed for the virtual server, we will see the first 128 bytes (-s128) in ASCII (-X).
   b. Go to your browser and attempt to access the virtual server. You should see something like this:

```
17:38:40.051122 IP 10.1.10.1.43932 > 10.1.10.235.http: S 522853636:522853636(0) win 8192 <mss 1460,nop,wscale 2,nop,nop,sackOK>
    0x0000:  0ffe 0800 4500 0034 0a40 4000 4006 0699  ....E..4.@@.@...
    0x0010:  0a80 0a01 0a80 0aeb ab9c 0050 1f2a 1d04  ...........P.*..
    0x0020:  0000 0000 8002 2000 3d10 0000 0204 05b4  ........=.......
    0x0030:  0103 0302 0101 0402                      ........
17:38:40.051169 IP 10.1.10.235.http > 10.1.10.1.43932: S 245310500:245310500(0) ack 522853637 win 4380 <mss 1460,sackOK,eol>
    0x0000:  0ffe 0800 4500 0030 27ef 4000 ff06 29ed  ....E..0'.@...).
    0x0010:  0a80 0aeb 0a80 0a01 0050 ab9c 0e9f 2424  .........P....$$
    0x0020:  1f2a 1d05 7012 111c 2a0e 0000 0204 05b4  .*..p...*.......
    0x0030:  0402 0000                                ....
17:38:40.053644 IP 10.1.10.1.43932 > 10.1.10.235.http: . ack 1 win 64240
    0x0000:  0ffe 0800 4500 0028 0a41 4000 4006 06a4  ....E..(.A@.@...
    0x0010:  0a80 0a01 0a80 0aeb ab9c 0050 1f2a 1d05  ...........P.*..
    0x0020:  0e9f 2425 5010 faf0 7018 0000            ..$%P...p...
17:38:40.053648 IP 10.1.10.1.43932 > 10.1.10.235.http: P 1:416(415) ack 1 win 64240
    0x0000:  0ffe 0800 4500 01c7 0a42 4000 4006 0504  ....E....B@.@...
    0x0010:  0a80 0a01 0a80 0aeb ab9c 0050 1f2a 1d05  ...........P.*..
    0x0020:  0e9f 2425 5018 faf0 43c5 0000 4745 5420  ..$%P...C...GET.
    0x0030:  2f20 4854 5450 2f31 2e31 0d0a 486f 7374  /.HTTP/1.1..Host
    0x0040:  3a20 3130 2e31 3238 2e31 302e 3233 350d  :.10.1.10.235.
```

   c. Well you are hitting the virtual server so let's look a little deeper and expand our dump. Your original client IP is in the first line of the dump 16:44:58.801250 IP **10.1.10.1**.41536 > 10.1.10.235.https:

3. In the second SSH window we will do an expanded **tcpdump** for the sake of interest.
   a. tcpdump –i <server vlan name> -X –s128 host <client IP>
   b. Hit your virtual server again. As you can see, we are sending packers to the pool members. They just aren't responding. So we can reasonably suspect it's a server issue.

4. It could be a port issue, let's check to see if the server is responding on port 80. On the BIG-IP in an SSH window:
    a. Do a **<ctrl-c>** to escape out of **tcpdump**, if you are still in it, and use **curl** to test the server.
        i. **curl –i <server ip>**
            1. "-i" dump the HTTP header information also.

```
[root@bigip249:Active:Standalone] config # curl -i 10.1.20.11
HTTP/1.1 200 OK
Date: Sat, 26 Jul 2014 19:25:28 GMT
Server: Apache/2.2.22 (Ubuntu)
X-Powered-By: PHP/5.4.9-4ubuntu2.2
Vary: Accept-Encoding
Content-Length: 3819
Connection: close
Content-Type: text/html

<html>
<head>
<TITLE>Using virtual server 10.1.20.11 and pool member 10.1.20.11 (Node #1)</TITLE>
<meta http-equiv="Content-Type" content="text/html; charset=us-ascii" />
```

    b. The server is responding to the BIG-IP when directly connected, but not through the virtual server. Sounds like the server is routing around the BIG-IP, which means the BIG-IP is not the default gateway.

Turn **SNAT Automap** back on the **www_vs** virtual server

LAB 7: DEVICE SERVICE CLUSTERS (DSC)

To familiarize you with the concept of Device and Traffic Groups as well as the building of Active-Standby, Active-Active BIG-IP pairs. While there is a wizard, for this lab configuration will be done manually. The wizard will only build A/S HA groups, to build Active-Active and beyond a pair you will need to know the four steps to add a device object to a cluster. This lab is based on the Device/Traffic Group lab in the V11 update course.

Base Networking and HA VLAN

You will be creating a high availability cluster using the second BIG-IP (**bigip2)** in your lab , so let's prep our current BIG-IP and we will be creating a high availability VLAN.

1. On **bigip01.f5demo.com** archive your configuration in case you need to revert.
    j. Go to **System > Archives** and create a new archive.
    k. You will be using your third interface (1.3) for Network Failover and ConfigSync. This requires certain ports to be open on the Self IP; TCP port 4353 for ConfigSync and TCP port 1026 for Network Failover and TCP port 6699 for the Master Control Program.
        i. Build a new untagged VLAN **ha_vlan** on interface **1.3**
        ii. Add a self-IP address to the VLAN, **10.1.30.245** net mask **255.255.255.0.**
            1. Under **Port Lockdown**, select **Allow Default**, to open ports required for HA communications.
                a. Optionally you could select: **Allow Custom** and add TCP ports 4353,1026 and 6699

2. Go to **https://10.1.1.246** which is **bigip02.f5demo.com** and login with the credentials **admin/f5UDFrocks!**.
    a. **bigip02** has already been licensed and provisioned. You will need to set up the base networking.

Interface	Untagged VLAN	Self IP	Netmask
1.1	client_vlan	10.1.10.246	255.255.255.0
1.2	server_vlan	10.1.20.246	255.255.255.0
1.3	ha_vlan	10.1.30.246	255.255.255.0

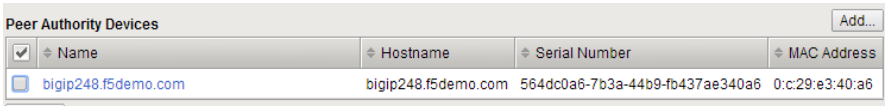
    b. On the **ha_vlan** ip configure set **Port Lockdown** to **Allow Default**
    c. Build the default gateway destination **0.0.0.0**, mask **0.0.0.0**, gateway ip address **10.1.10.1**
    d. What is the status your BIG-IPs? Check the upper left-hand corner next to the F5 ball.

Configure HA

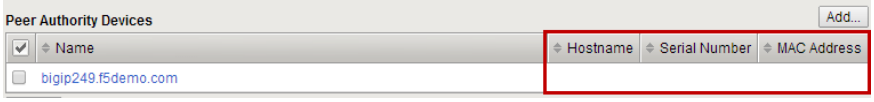
1. **On each BIG-IP**, prior to building the Device Trust it is recommended renewing the BIG-IP self-signed certificate with valid information and re-generating the local Device Trust certificate.
    a. Under **System > Device Certificate > Device Certificate** select the **Renew…** button
        i. **Common Name**: <the Hostname of the BIG-IP in the upper left corner>
        ii. **Country**: United States (or your country of preference)
        iii. **Lifetime**: 3650
            1. Lifetime is important, if your cert expires your HA setup will fail.
        iv. Select **Finished**. Your browser will ask to exchange certs with the BIG-IP again.
    b. Under **Device Management > Device Trust > Local Domain** select **Reset Device Trust…**
        i. In the **Certificate Signing Authority** select **Generate New Self-Signed Authority** and hit **Update**.

2. **On each BIG-IP** configure the device object failover parameters the BIG-IP will send to other BIG-IPs that want to be a part of a sync-only or sync-failover group.
    a. Under **Device Management>Device**, select the local BIG-IP. It will have the **(Self)** suffix.
        i. Under **Device Connectivity** on the top bar select:
            1. **ConfigSync**
            2. Use the Self IP address of the HA VLAN for your **Local Address**.
        ii. **Network Failover**
            1. In the **Failover Unicast Configuration** section select the **Add** button
            2. Use the Self IP address the HA VLAN for your **Address**
            3. Leave the **Port** at the default setting of 1026
            4. **Note:** Multicast is for Viprion chasses only.
        v. **Mirroring**
            1. **Primary Local Mirror Address**: use the Self IP address of the HA VLAN for your
            2. **Secondary Local Mirror Address:** None

7. On **bigip01.f5demo.com** build the Device Trust.
    a. Under **Device Management>Device Trust> Device Trust Members** and select **Add** to add other BIG-IP(s) you will trust.
        i. **Device IP Address**: <management IP address of the BIG-IP to add>
            1. You could use any Self IP if the out-of-band management interface is not configured.
        ii. Enter the Administrator Username and Password of the BIG-IP you are trusting.
        iii. Select **Retrieve Device Information**
            1. The certificate information and name from the other BIG-IP should appear
            2. Select **Device Certificate Matches**
        iv. Select **Add Device**.
            1. On each BIG-IP check the other BIG-IP in the **Peer Authorities** list. ***Is all the information there?***



v. If some information is missing delete the trust and try again.



vi. What are the statuses of your BIG-IPs now?
   1. They should be **In Sync**. But wait! We haven't even created a device group! But remember the **Device Trust** creates a **Sync-Only** group for the certificates under the covers (*device-trust-group*) and that should be in sync.
   2. Click on **In Sync** in the upper right corner or **Device Management>Overview** to see the **device_trust_group**.

8. On bigip01.f5demo.com create a new **Sync-Failover** device group
   a. **Under Device Management>Device Groups** create a new device group.
      i. **Name:** my-device-group
      ii. **Group Type**: Sync-Failover
      iii. Add the members of the group to the **Includes** box and select **Finished**.
      iv. Check **Device Groups** on each BIG-IP.
      v. Did you have to create the Device Group on the other BIG-IP?
         1. Is the full configuration synchronized yet? (No! Only the Device Group is sync'd)
      vi. What is your sync status?
         1. It should be **Awaiting Initial Sync**
      vii. Click on the sync status or go to **Device Management>Overview** (or click on **Awaiting Initial** Sync) of the BIG-IP with the **good/current** configuration.
      viii. Click the device with the configuration you want to synchronize. **Sync Options** should appear.
      ix. **Push the selected device configuration to the group**. It could take up to 30 seconds for synchronization to complete.
         1. What are the statuses of your BIG-IPs? Do you have an active-standby pair?
         2. Are the configurations the same?

9. Now that you have created your HA environment. HA selections will show up for SNAT addressed (not tied to your base network), persistence profiles and connection mirroring on virtual servers.
   e. Go to your **Active** BIG-IP.
   f. Go to your persistence profile **my-src-persistence** and check the **Mirror Persistence** box.
   g. Go to your **www_vs** virtual server and set the **Default Persistence Profile** to **my-src-persistence.**
   h. Synchronize your changes. Did the changes sync?
   i. On each BIG-IP go to **Module Statistics > Local Traffic** and bring up the persistence record statistics.
      i. Go to the home page of you www_vs web service (http://10.1.10.100). Refresh a few times.
      ii. Check the persistence records on each of your BIG-IPs, you should see the records are mirrored on each device.

3. Go to **Device Management>Traffic Groups**. As you can see the default traffic group "**traffic-group-1**" already exists.
   a. Select **traffic-group-1**, check out the page information and then select **Force to Standby**.
   b. What are the statuses of your BIG-IPs? Go to your web page. What is the client IP?
   c. Go to your self-IP addresses. What traffic group are they in? What does it mean?
   d. Archive your work.

BONUS LAB – TRAFFIC GROUPS, IAPPS AND ACTIVE-ACTIVE

If you have time this is a bonus lab. Here you will create a new traffic group. You will use iApps to an create a new HTTP application that resides in that address group and you will create a floating IP address that will be used as the default gateway that also resides in that traffic group.

Building a new traffic group and floating IP.

1. On you **Active** BIG-IP, go to **Device Management>Traffic Groups** and hit Create.
    a. Use the f5.http template, which was designed for general web services
        i. **Name**: iapp_tg
        ii. Take the defaults for the rest.

2. Add a floating SelfIP to the **server_vlan**. Go to **Network>Self IP**
    b. **Name:** server_gateway
    c. **IP Address:**10.1.20.240
    d. **Netmask:** 255.255.255.0
    e. **VLAN/Tunnel:** server_vlan
    f. **Traffic Group:** iapp_tg (floating)

Building an HTTP application using an iApp template.

1. Go to **iApp>Application Services** and hit **Create**.
    a. Use the f5.http template, which was designed for general web services
        i. Set the **Template Selection** to **Advanced.**
        ii. **Name**: my_new_iapp
        iii. **Traffic Group:** iapp_tg (floating)
            1. You will have to uncheck the **Inherit traffic group from current partition / path**.
        iv. Under **Template Options**
            1. Select the **Advanced – Configure advanced options** for the configuration mode.
        v. Under **Network**
            1. **How have you configured routing on your web servers?:** Servers have a route to the clients through the BIG-IP system.
                a. In other words, the BIG-IP is the default gateway for the servers.
                b. Otherwise the template would use SNAT by default.
        vi. Under **Virtual Server and Pools**
            1. Your virtual server IP is **10.1.10.110**
            2. Your hostname will be **www.f5demo.com** because you have to put one in.
            3. Create a new pool with the members **10.1.20.14:80** and **10.1.20.15:80**
                c. **If you hit add after the last pool member and have a new row, you will need to delete the row prior to finishing.**

vii. Hit **Finished** at the bottom of the page.

2. Go to **iApp>Application Services** and select the new application you created.
   a. Select **Components** from the top bar.
      i. Here you will see all the configuration items created by the iApp
      ii. Do you see anything created that you weren't asked about?

3. Remember the concept of strictness? Let's test that out`
   a. Go to Local Traffic>>Pools>>Pool List
      i. Select the pool created by your iApp **my_new_iapp_pool**
      ii. Attempt to add **10.1.20.13:80** to your **my_new_iapp_pool**.
         2. Did it fail?
   b. Go to your iApp and select Reconfigure from the top bar.
      i. Now attempt to add your new pool member.
      ii. You can check the Components tab to verify your success.

SYNCHRONIZE YOUR CHANGES

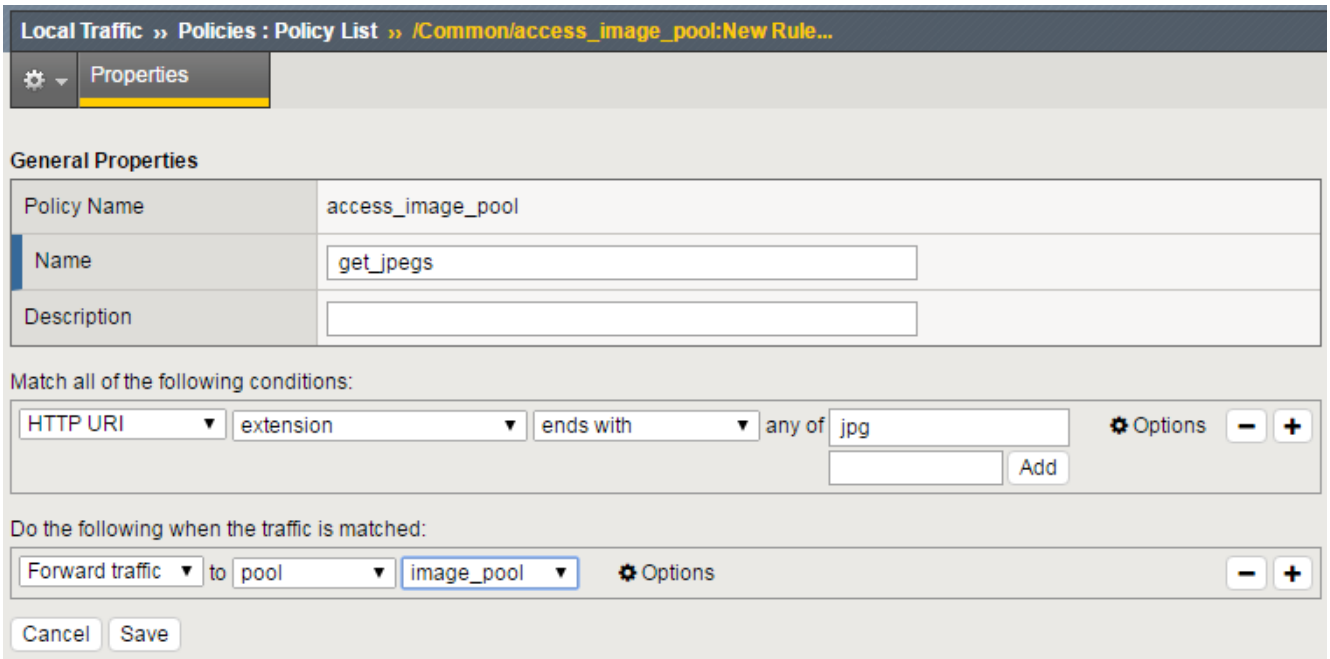
Active-Active Setup

1. Now, let's make our sync-failover group active-active. On the **Active** BIG-IP:
   a. Go to **Device Management > Traffic Groups**
      i. Go to you **iapp_tg** traffic group.
      ii. Under **Advanced Setup Options**
         1. You are going to set up **iapp_tg** to prefer to run on **bigip02.f5demo.com** and auto failback to **bigip02** if **bigip02** should go down and come back up later.
         3. Is this normally a good idea?
      iii. **Failover Method:** HA Order
      iv. **Auto Failback:** <checked>
      v. **Failover Order: bigip02.f5demo.com**then **bigip01.f5demo.com**
      vi. Ensure you synchronized the change to the other BIG-IP.

10. If the traffic group is active on the wrong BIG-IP initially you will have to do a Force to Standby on the traffic group to make it active on BIG-IP you want it on by default.
    a. What is the ONLINE status of each of your BIG-IPs
    b. Reboot the BIG-IP with your second traffic group on it. Watch to see if the becomes active on other BIG-IP during the reboot and if it falls back to the Default Device once the BIG-IP has come back up.
    c. You can verify this by checking your traffic groups or going to the web server and looking at the client IP.

APPENDIX

BIG-IP Policy for retrieving jpeg images from the image_pool



BIG-IP extra credit iRule to add PNG to access_image_pool iRule