Achieving High Availability with
Websphere Application Server SIP Container and F5 BIG-IP Local Traffic Manager

Ollie J. Hales, ojhales@us.ibm.com
Tamera L. Davis, tameraj@us.ibm.com
Tibor Beres, tberes@us.ibm.com
Nojan Moshiri, N.Moshiri@f5.com

Summary:

This paper is designed to outline the configuration and deployment of a commonly used configuration of the WebSphere Application Server SIP Container - the High-Availability (HA) environment. This HA environment will be configured using the F5 BIG-IP LTM product.

This topic is important because WebSphere SIP HA environments are becoming more popular among SIP deployments. Many companies are using the F5 BIG-IP already in-house and so this paper will answer many questions about the usage of the BIG-IP LTM with IBM SIP container. So this article will be of great benefit to administrators on how to setup and configure this environment using both products.

This article is designed for system administrators and also those just getting started with WebSphere SIP Container.

Outline

I. Introduction to IBM WebSphere SIP Container HA Environment

II. Deploying a SIP Configuration - Dual Proxy High Availability (HA) Environment with F5 BIG-IP LTM
   a. WebSphere SIP Proxy Server Configuration
   b. F5 BIG-IP Configuration

III. Running SIP Traffic
   a. Using the UDP Protocol
   b. Using the TCP Protocol
      i. Using the TCP Outbound Feature
   c. Using the TLS Protocol
   d. Using Converged (SIP/HTTP) Applications

IV. Conclusion
I. Introduction to IBM WebSphere SIP Container HA Environment

Highly Available (HA) environments are very popular among IBM WebSphere SIP deployments. This will ensure that there is no single point of failure, especially in environments where a Load Balancer is in use for proxy server high-availability in addition to application server HA.

For environments requiring a high availability environment, replication domains are configured to enable memory-to-memory replication between the SIP containers. When using WebSphere Application Server ND, it is advised that no more than two application servers be included in a replication domain, due to the sheer volume of data that is replicated in a SIP session.

With memory-to-memory replication enabled, the established SIP Application Sessions are replicated to its domain partner, therefore, when one member fails, all established sessions can be handled by the partner server. The following figure below shows how such an environment requiring high availability can be configured using F5 BIG-IP LTM and two proxy servers. In this example, IBM Blade chassis’ are in use, with both the first and second chassis housing four blade servers. In the first chassis, the WebSphere ND Deployment Manager runs on the same blade as the SIP Proxy. The remaining three blades contain two application servers each. In the second chassis, the first blade contains a proxy server and the remaining three blades contain two application servers each.

Many companies are using the F5 BIG-IP already in-house and so this paper will answer many questions about the usage of the BIG-IP with IBM SIP container. So this article will be of great benefit to administrators on how to setup and configure this environment using both products.

Fig. 1
II. Deploying a SIP Configuration - Dual Proxy High Availability (HA) Environment with F5 BIG-IP

a. WebSphere SIP Proxy Server Configuration

This document assumes that the system administrator has followed the instructions in document [Configuring and Deploying WebSphere SIP Environments] and has created a standard WebSphere cell containing 1 proxy server and 8 application servers.

In order for your proxy server to correctly interoperate with the F5 BIG-IP Local Traffic Manager (LTM), the proxy server must be configured to accept and handle all incoming packets from the F5 BIG-IP LTM device. To do this, the following changes need to be made on the proxy server panels on your WebSphere Administration Console.

1.) Enable Load Balancer Health Checking

In order to keep track of the availability of a backend proxy server, the F5 BIG-IP LTM sends a stream of SIP OPTIONS messages at regular intervals to any proxy server it is routing client traffic to. The proxy in turn is expected to respond to these option messages with a 200 OK. This [send OPTIONS \(\rightarrow\) \(\leftarrow\) expect 200 OK] handshake mechanism is what allows the F5 BIG-IP LTM to query the health status of its backend proxy servers. If the proxy does not respond within a pre-determined time frame, the F5 BIG-IP LTM will interpret...
the lack of response as an acknowledgement of an inactive proxy server. The F5 BIG-IP LTM will then discontinue forwarding any client traffic to the downed server. The F5 BIG-IP LTM, will however, continue sending OPTIONS requests to the backend proxy server and wait for a 200 OK response. At the point in time it receives a 200 OK response, the F5 BIG-IP LTM will then resume forwarding client packets to the backend proxy server.

For this mechanism to work, the WebSphere SIP Proxy must be configured to expect OPTIONS messages from the F5 BIG-IP LTM. To configure the proxy server to accept OPTIONS messages from F5 BIG-IP LTM, browse to the following location on the WebSphere Administration console:

Servers → Server Types → WebSphere Proxy Servers → [Your SIP Proxy Server Name] → SIP Proxy Server Settings → SIP Proxy Settings

Browse down the panel and look for the heading: **Load Balancer Health Checking**

There are 2 fields that you will need to complete.

**IP address 1:**
Specify the physical IP address of the Load Balancer that sources the SIP health checks.

**SIP health check method name:**
This string value represents the method name that the Load Balancer is sending to the SIP Proxy, which is used for health checks.

In the above illustration, the IP address 9.37.30.210 represents the self-ip address of our F5 BIG-IP LTM. The BIG-IP self-ip is configured during the network setup portion of the LTM and monitoring traffic originates from this IP address. We also have OPTIONS specified as the SIP health check method name. Based on what is configured, this SIP Proxy server will now accept all OPTIONS messages that are determined to have a source address of 9.37.30.210 and reply to them with a 200 OK response.

**2.) Enable Proxy Specific Custom Properties**

There are a set of SIP proxy specific custom properties that must also be present and configured before interoperability between the F5 BIG-IP LTM and the WebSphere SIP proxy server. To set these customer properties, browse to the following location within the WebSphere application server:

Servers → Server Types → WebSphere Proxy Servers → [Your SIP Proxy Server Name] → SIP Proxy Server Settings → SIP Proxy Settings → Custom Properties

Add the following custom properties:

a.) forceRport = true

- forceRport will add an "ibmrport" to the incoming Via header. We'll use the port number to map the response to the correct client connection
b.) useViaSentByForOutboundConnections = true
   - useViaSentByForOutboundConnections tells the proxy to use the IP/hostname in the sent-by parameter to make outbound connections to the client. By default, it will use the address in the received parameter.

c.) udp.IPSprayer.host = IP address of the cluster
   - IP address of the cluster from which the proxy server is expecting to receive UDP packets from the F5. This address corresponds to the Self-IP Address that the administrator configured on the F5 BIG-IP LTM.

d.) udp.IPSprayer.port = Port address of the cluster
   - Port address of the cluster from which the proxy server is expecting to receive UDP packets from the F5. If “port” property is not set, it defaults to port 5060

e.) tcp.IPSprayer.host = IP address of the cluster
   - IP address of the cluster from which the proxy server is expecting to receive TCP packets from the F5. This address corresponds to the Self-IP Address that the administrator configured on the F5 BIG-IP LTM.

f.) tcp.IPSprayer.port = Port address of the cluster
   - Port address of the cluster from which the proxy server is expecting to receive TCP packets from the F5. If “port” property is not set, it defaults to port 5060

g.) tls.IPSprayer.host = IP address of the cluster
   - IP address of the cluster from which the proxy server is expecting to receive TLS packets from the F5. This address corresponds to the Self-IP Address that the administrator configured on the F5 BIG-IP LTM.

h.) tls.IPSprayer.port = Port address of the cluster
   - Port address of the cluster from which the proxy server is expecting to receive TLS packets from the F5. If “port” property is not set, it defaults to port 5061

i.) ipForwardingLBEnabled = true
   - This custom property enables all custom IP forwarding code and must be enabled for IP forwarding to work.
b. F5 BIG-IP Configuration

SIP Configuration for BIG-IP

This section of the document will detail instructions as to how to configure the BIG-IP to work with the WebSphere SIP Proxy Server. This section of the document makes the following assumptions. It assumes that the administrator has already configured all network interfaces, VLAN(s), Self-IP addresses, and routes needed for your BIG-IP LTM to function.

The administrator will begin the configuration process updating the Virtual Server configuration. The Virtual Server configuration consists of four main actions:

1. Creation of profiles
2. Creation of monitors
3. Creation of pools
4. Creation of virtual servers

**Action 1: Creation of profiles**

Profiles are the components that allow for the customization of settings on a virtual server. Profiles provide optimization, SIP intelligence and SIP control for traffic passing through the BIG-IP. We will create seven profiles, six via the Graphical User Interface (GUI) and one using the command line. The profiles are:

- SIP Profile - Configures the BIG-IP to participate as a SIP Proxy
- SIP Persistence profile - Configures the BIG-IP to handle persistence for SIP sessions
- A TCP Profile - Configures optimization of TCP connections through the BIG-IP
- A UDP profile - Configures optimization of UDP connections through the BIG-IP
- A Message Based Load balancing (MBLB) profiles - Configures the BIG-IP to be able to buffer and optimize message based traffic (via the command line)
- A client SSL profile - Configures the termination of TLS traffic from the client to the BIG-IP
- A server SSL profile - Configures the initiation of TLS traffic from the BIG-IP to the Websphere SIP Proxy

SIP Profile creation

To create the SIP profile

1. On the Main tab, expand Local Traffic, and then click Profiles.
2. On the Menu bar, from the Services menu, click SIP.
3. Click the Create button.
4. In the Name box, type a name for this profile. In our example, we type SIP-profile. Configuring the BIG-IP LTM for SIP Traffic Management
5. From the Insert Via Header row, click the Custom box, and then select Enabled from the list.
6. From the Terminate-on-bye row, click the Custom box, and then disable this option
7. Modify any of the settings as applicable for your network. Refer to the online help for details.
8. In our example, we leave the settings at their default levels.
9. Click the Finished button.
SIP Persistence Profile

To create the SIP persistence profile

1. On the Main tab, expand Local Traffic, and then click Profiles.
2. On the Menu bar, click Persistence.
3. Click the Create button.
4. In the Name box, type a name. We type SIP-persistence.
5. From the Persistence Type list, select SIP.
6. From the SIP-Info row click the Custom box and then select Call-ID
7. Click on the Custom box in the Timeout row and set a Timeout value of 180 seconds
8. All other settings are optional; configure as applicable for your implementation.
9. Click the Finished button.
A TCP Profile

To create the TCP profile

1. On the Main tab, expand Local Traffic, and then click Profiles.
2. On the Menu bar, from the Protocol menu, click TCP.
3. Click the Create button.

4. In the Name box, type a name for this profile. In our example, we type SIP-TCP-profile.
5. Optional: In the Idle Timeout row, configure an appropriate timeout.
6. Configure any of the other options as applicable for your implementation. In our example, we leave the settings at the default levels.
7. Click the Finished button.
A UDP Profile

To create the UDP profile

1. On the Main tab, expand Local Traffic, and then click Profiles.
2. On the Menu bar, from the Protocol menu, click UDP.
3. Click the Create button.

4. In the Name box, type a name for this profile. In our example, we type SIP-UDP-profile. (To be consistent?)
5. Optional: In the Idle Timeout row, configure an appropriate timeout.
   Configure any of the other options as applicable for your implementation. In our example, we leave the settings at the default levels.
6. Click the Finished button.
Message Based Load Balancing Profile (MBLB)

The MBLB profile has to be created and applied via the command line, we do not provide GUI support for it yet. The instructions are as follows:

1. Login via SSH to your BIG-IP, as root
2. Enter the tmsh shell by entering tmsh at the command prompt then pressing enter:
   * tmsh
3. Go to the LTM profile section of the tmsh by entering ltm, enter then profile, enter:
   * ltm
   * profile
4. Create the MBLB profile by entering the command create mblb websphereMBLBprofile, then enter:
   * create mblb websphereMBLBprofile

```
[moshiri@SEA-L-MOSHIRI01 ~]$ ssh root@192.168.17.50
Password: Last login: Wed Jan 18 16:00:44 2012 from 192.168.17.1
[root@bigipmoshiri2(Active) config # tmsh
root@bigipmoshiri2(Active) [/Common]{tmos}# ltm
root@bigipmoshiri2(Active) [/Common]{tmos.ltm}# profile
root@bigipmoshiri2(Active) [/Common]{tmos.ltm.profile}# create mblb websphereMBLBprofile]
```

A client SSL profile

The creation of a client SSL profile allows for the termination of SSL traffic on the BIG-IP. This offloads SSL processing from the end servers. This is a two step process which includes the creation of the key and signed certificate (or importing of the key and certificate) and then creation of the profile.

To create or import a new Client SSL certificate

1. On the Main tab, expand Local Traffic, click SSL Certificate list, and then select either Import or Create
2. If you select Import select the key for your certificate and the signed certificate  
3. If you select Create, you can create a self-signed certificate

To create a new Client SSL profile

1. On the Main tab, expand Local Traffic, click Profiles, and then, on the Menu bar, from the SSL menu, select Client  
2. Click the Create button.
3. In the Name box, type a name for this profile. In our example, we type SIP-client-ssl.
4. In the Configuration section, click a check in the Certificate and Key Custom boxes.
5. From the Certificate list, select the appropriate Certificate
6. From the Key list, select the appropriate Key.
7. Click the Finished button.

A server SSL profile

If security requires that all traffic within the network be secure, a Server SSL profile must be used to re-encrypt traffic back to the back end proxy servers. This is a two step process involving the creation of the key and signed certificate (or importing of the key and certificate) and then creation of the profile.

To create or import a new Client SSL certificate
1. On the Main tab, expand Local Traffic, click SSL Certificate list, and then select either Import or Create
2. If you select Import, import the key for your certificate and the signed certificate
3. If you select Create, you can create a self-signed certificate

To create a new Server SSL profile

1. On the Main tab, expand Local Traffic, click Profiles, and then, on the Menu bar, from the SSL menu, select Server.
2. Click the Create button.

3. In the Name box, type a name for this profile. In our example, we type SIP-server-ssl.
4. In the Configuration section, click a check in the Certificate and Key custom boxes.
5. From the Certificate list, select the appropriate Certificate
6. From the Key list, select the appropriate Key.
7. Click the Finished button.
Action 2: Creation of monitors

To create the SIP monitor

1. On the Main tab, expand Local Traffic, and then click Monitors.
2. Click the Create button. The New Monitor screen opens.
3. In the Name box, type a name for the Monitor. In our example, we type SIP-monitor-UDP.
4. From the Type list, select SIP. The SIP monitor options appear.
5. Configure any of the other options as applicable for your implementation. In our example, we choose UDP and we leave the default settings.
6. Click the Finished button.
7. Repeat this process for TCP, changing the name to SIP-monitor-TCP and create a TCP Monitor instead of the UDP monitor.
Action 3: Creation of pools

Two pools will be created, one for TLS and one for Non-TLS.

- Create a NON-TLS pool for port 5060 traffic
- Create a TLS pool for port 5061 traffic

Create the SIP POOL

In this section we will create the pools where SIP traffic will be delivered. We will create a total of three pools, one for UDP, one for TCP and one for TLS. Each pool will have the appropriate combination of port number and monitor.

To create the SIP UDP pool

1. On the Main tab, expand Local Traffic, and then click Pools.
2. Click the Create button. The New Pool screen opens.
3. In the Name box, type a name. In our example, we use SIP-UDP-pool.
4. In the Health Monitors section, select the name of the monitor you created in Creating the SIP UDP monitor, and then click the Add (<<) button. In our example, we select SIP-monitor.
5. From the Load Balancing Method list, select Round Robin.
6. For this pool, we leave the Priority Group Activation Disabled.
7. In the New Members section, in the Address box, add the first server to the pool. In our example, we type 192.0.2.123
8. In the Service Port box, type the service number you want to use for this device, or specify a service by choosing a service name from the list. In our example, we type 5060.
9. Click the Add button to add the member to the list.
10. Repeat steps 6-8 for each device you want to add to the pool.
11. Click the Finished button
Create the SIP TLS Pool

1. On the Main tab, expand Local Traffic, and then click Pools.
2. Click the Create button. The New Pool screen opens.
3. In the Name box, type a name. In our example, we use SIP-TLS-pool.
4. In the Health Monitors section, select the name of the monitor you created in Creating the SIP monitor, and then click the Add (<<) button. In our example, we select SIP-monitor.
5. From the Load Balancing Method list, select Round Robin.
6. For this pool, we leave the Priority Group Activation Disabled.
7. In the New Members section, in the Address box, add the first server to the pool. In our example, we type 192.0.2.123
8. In the Service Port box, type the service number you want to use for this device, or specify a service by choosing a service name from the list. In our example, we type 5061.
9. Click the Add button to add the member to the list.
10. Repeat steps 6-8 for each device you want to add to the pool.
11. Click the Finished button

Create the SIP TCP Pool

1. On the Main tab, expand Local Traffic, and then click Pools.
2. Click the Create button. The New Pool screen opens.
3. In the Name box, type a name. In our example, we use SIP-TCP-pool.
4. In the Health Monitors section, select the name of the SIP TCP monitor you created in Creating the SIP monitor, and then click the Add (<<) button. In our example, we select SIP-TCP-monitor.
5. From the Load Balancing Method list, select Round Robin.
6. For this pool, we leave the Priority Group Activation Disabled.
7. In the New Members section, in the Address box, add the first server to the pool. In our example, we type 192.0.2.123
8. In the Service Port box, type the service number you want to use for this device, or specify a service by choosing a service name from the list. In our example, we type 5061.
9. Click the Add button to add the member to the list.
10. Repeat steps 6-8 for each device you want to add to the pool.
11. Click the Finished button

Action 4: Creation of Virtual Servers

We will create 3 virtual servers, one for TCP traffic, one for UDP Traffic, and finally one for encrypted TLS traffic for TCP

- TCP Traffic
- UDP Traffic
- TLS traffic for TCP

Create the virtual server for TCP traffic

We will now create Virtual Servers that bring together all of the profiles and pools that we created in the steps above. Virtual Servers can have the same IP address but listen on different ports and for different protocols. In our examples we will be creating three virtual servers, one for UDP on port 5060, one for TCP on port 5060 and one for TLS on port 5061.

To create the TCP Virtual Server

1. On the Main tab, expand Local Traffic, and then click Virtual Servers.
2. Click the Create button.

3. In the Name box, type a name. We type SIP-virtual-TCP.
4. In the Destination section, select the Host option button.
5. In the Address box, type the IP address of this virtual server. In our example, we type 10.133.81.22.
6. In the Service Port box, type 5060.
7. From the Configuration list, select Advanced.

8. From the Protocol list, select TCP.
9. From the Protocol Profile (Client) list, select the profile you created in Creating the TCP profile. In our example, we type SIP-TCP-profile.
10. From the SIP Profile list, select the profile you created Creating the SIP profile. In our example, we select SIP-profile.
11. From the Default Pool list, select the TCP pool you created in Creating the SIP load balancing pool. In our example, we select SIP-pool.
12. From the Default Persistence Profile list, select the SIP persistence profile you created in Creating a SIP persistence profile. In our example, we type SIP-persistence.
13. Click the Finished button.

Create the virtual server for UDP traffic
To create the virtual server

1. On the Main tab, expand Local Traffic, and then click Virtual Servers.
2. Click the Create button.
3. In the Name box, type a name. We type SIP-virtual-UDP.
4. In the Destination section, select the Host option button.
5. In the Address box, type the IP address of this virtual server. In our example, we type 10.133.81.22.
6. In the Service Port box, type 5060.
7. From the Configuration list, select Advanced.
8. From the Protocol list, select UDP.
9. From the Protocol Profile (Client) list, select the profile you created in Creating the UDP profile. In our example, we type SIP-UDP-profile.
10. From the SIP Profile list, select the profile you created Creating the SIP profile. In our example, we select SIP-profile.
11. From the Default Pool list, select the UDP pool you created in Creating the SIP load balancing pool. In our example, we select SIP-pool.
12. From the Default Persistence Profile list, select the SIP persistence profile you created in Creating a SIP persistence profile. In our example, we type SIP-persistence.
13. Click the Finished button.

Create the virtual server for TLS traffic over TCP

To create the virtual server
1. On the Main tab, expand Local Traffic, and then click Virtual Servers.
2. Click the Create button.
3. In the Name box, type a name. We type SIP-virtual-TLS.
4. In the Destination section, select the Host option button.
5. In the Address box, type the IP address of this virtual server. In our example, we type 10.133.81.22.
6. In the Service Port box, type 5061.
7. From the Configuration list, select Advanced.
8. From the Protocol list, select TCP.
9. From the Protocol Profile (Client) list, select the profile you created in Creating the TCP profile. In our example, we select SIP-TCP-profile.
10. From the SSL Profile (Client) list, select the profile you created using the SSL Client profile creation.
11. From the SSL Profile (Server) list, select the profile you created using the SSL Server profile creation.
12. From the SIP Profile list, select the profile you created Creating the SIP profile. In our example, we select SIP-profile.
13. From the Default Pool list, select the TLS pool you created in Creating the SIP load balancing pool. In our example, we select SIP-TLS-pool.

14. From the Default Persistence Profile list, select the SIP persistence profile you created in Creating a SIP persistence profile. In our example, we type SIP-persistence.

15. Click the Finished button.

Now, from the command line, we have to associate the MBLB profile we created earlier with the virtual servers we created above.

1. Login via SSH to your BIG-IP, as root.
2. Enter the tmsh shell by entering tmsh at the command prompt then pressing enter:
   * tmsh
3. switch to the ltm virtual area of tmsh by entering /ltm virtual (not the forward slash), and then enter:
   * /ltm virtual
4. Enter the configuration of your virtual server by entering modify <your virtual server name>, and then enter, in this case, I will enter modify SIP-virtual-TLS, then enter:
modify SIP-virtual-TLS

5. Now add the MBLB profile to the virtual server by using the profiles add command:

   * profiles add { websphereMBLBprofile }

6. Now save your configuration by using the save /sys config command (note the slash), and press enter:

   * save /sys config

Repeat the above procedure for all of your virtuals, making sure you save /sys config at the end.

III. Running SIP Traffic

This section details the various specific configuration changes that need to be made to the F5 BIG-IP LTM server configuratoin to accommodate different traffic protocol when running SIP traffic to the WebSphere SIP Proxy Server.

Using the UDP Protocol

No specific configuration changed are required.

Using the TCP Protocol

No specific configuration changed are required.

Using the TCP Outbound Feature

The SIP feature RFC 5626: Managing Client-Initiated Connections in the Session Initiation Protocol is supported by WebSphere Application Server 7.0.0.21 and above. However, to enable this feature when using the F5 BIG-IP LTM, the administrator MUST DISABLE the use of the SIP profile on the virtual server configuration tab.

Reason: The SIP Profile strips off the flow token in the Record Route request required by RFC 5626. Subsequent requests that need to be routed back to the caller using this flow token are unable to due to the fact it is missing on the initial request. To work around this issue, we disable the SIP Profile.

1. On the F5 BIG-IP LTE, follow this path: Local Traffic ➔ Virtual Servers ➔ Virtual Servers List ➔ <name of your virtual server>
2. On the configurations drop down, click on advanced.
3. Scroll to the “SIP Profile” and select None.
4. Scroll to the bottom of the screen and click update

Using the TLS Protocol

In order to run TLS SIP traffic through F5 BIG-IP LTM you will have to have completed the following procedures below.

1. Configured a set of virtual servers for TLS over TCP. This procedure is defined in the section “Create the virtual servers for TLS traffic over TCP”.

2. Enable SNAT Automap

   If the server nodes (proxies, in the case of this document) do not have the BIG-IP as their default route, an asymmetric routing situation will occur. In the case of UDP traffic for SIP, BIG-IP participates as a SIP proxy and inserts its own VIA header, creating the proper routing chain with the downstream proxy. In some cases, especially with TCP traffic, simply inserting the VIA header may not be enough. Therefore, if the proxy does not have the BIG-IP as its default route, a SNAT should be used to rewrite the address so that the proxy responds directly back to the BIG-IP in return traffic.

   The benefit of using SNAT Automap is its ease of use. Simply turning on SNAT Automap in the Virtual server will use the Self-IP (as described in the paragraph above) and traffic will flow properly.

   The downside of use SNAT Automap is that by using the one Self-IP address, there is a limit of approximately 64,000 ports open at one time. If you are anticipating more than 64,000 simultaneous TCP connections, then a SNAT pool should be used instead of SNAT auto map.

SNAT POOLS

If you decide that you want to use a SNAT pool as the way to specify translation addresses in your SNAT, you must first create the SNAT pool, specifying one or more translation addresses that you want to include in the SNAT pool. You create a SNAT pool using the Configuration utility.

To create a SNAT pool

1. On the Main tab of the navigation pane, expand Local Traffic, and click SNATs. The SNATs screen opens.
2. On the menu bar, click **SNAT Pool List**. This displays a list of existing SNAT pools.

3. In the upper-right corner of the screen, click **Create**. Note: If the Create button is unavailable, this indicates that your user role does not grant you permission to create a SNAT.

4. For the **Name** setting, type a unique name for the SNAT pool.

5. For the **Member List** setting, type an IP address.

6. Click **Add**.

7. Repeat steps 5 and 6 for each translation address that you want to add.

8. Click **Finished**.

After the SNAT pool has been created you apply it to the Virtual Server just as you would apply a SNAT auto map.

3. **Configured the WebSphere Proxy’s security certificate and key management.**

   In order for the WebSphere SIP Proxy server to be ready to accept TLS packets from the F5 BIG-IP LTM, the SIP Proxy server’s SSL security certificate and key management must be configured to use the same public key that was imported into the F5 BIG-IP LTM configuration during the creation of both the server ssl profile and client ssl profile.

   To illustrate this process, this document will use the Redhat linux distribution of the free open source certificate authority Open SSL for the issuing and signing of keys.

   Configuring the WebSphere Environment will consist of the following action steps.

   1. Public Key Import from Certificate Authority.
   2. Creating a Personal Certificate Request.
   3. Signing the Personal Certificate Request using the Certificate Authority
   4. Importing the Signed Certificate into the WebSphere Environment.
   5. Configuring the WebSphere Environment’s DefaultTrustStore and DefaultKeyStores

   **Action Step 1: Public Key Import From Certificate Authority**

   1. Go to the directory where you installed your certificate authority. In the example below this directory is called /ca-bleached.
2. Locate the ca-cert.pem file and copy this file to the file system of your WebSphere Deployment Manager.

3. On the WebSphere Deployment Manager Console follow the following path: Security → SSL certificates and key management → Key Stores and Certificates.

4. On the keystores and certificates menu select CellDefaultTrustStore.
5. Next select Signer Certificates.

6. By default there will only be root and datapower certificates. The next step is to import the signer certificate that was just copied from the Open SSL certificate authority into the CellDefaultTrustStore. To do this, click add.
7. Name the certificate a name that can be associated with the certificate authority. Select the file name of the certificate that was copied over to the Deployment Manager host. Click OK.
8. Click Save.
9. The certificate has been successfully imported.

**Action Step 2: Creating a Personal Certificate Request**

The next step in the process is to get a WebSphere personal certificate signed by the OpenSSL certificate authority. To do this, a personal certificate request must be created.

1. On the WebSphere Administration Console, browse to Security → SSL certificate and key management → Key stores and certificates → Personal certificate requests

2. Select New.
3. Select a filename and export location for your certificate.
4. Complete all fields with information related to your environment.
5. Apply and Save those changes.
6. A personal certificate request will be created and exported to the location on your file system you specified in the creation process.

Action Step 3: Signing the Personal Certificate Request using the Certificate Authority

The next step involves moving the newly created personal certificate request from the WebSphere Deployment Manager’s file system to the file system of the Certificate Authority. Once there the personal certificate request can be signed.

1. Copy the file from the WebSphere Deployment Manager’s file system to the file system of the Certificate Authority.

```
[root@dev-r1c4b09 ~]# ls / 547767-patch.jar dmz-tunnel-2.0 lost+found @linux
bin stc media SIF
root final-signer-request memDumpLog SIF-JAMES
ca-bart-proxy-request final-signer-signed misc srv
ca-bart-proxy-signed final-template.props mnt sys
ca-bart-request gatherdata_cut.txt myNewCertRequest targetTree.xml
ca-bart-signed gatherdata.sh mmon_start test
ca-cert-bart-request home opt tftpboot
cert-signed IH post tmp
ca-cert.pem initrd proc tune.sh
cell-cert-request lib-alias.sh Production8-23a-patch.jar tuning2-script
cell-cert-signed lib root tuning-script-
dev load-balancer root tuning-scripts
[root@dev-r1c4b09 ~]# scp /myNewCertRequest svt-r1c1b01:/ca-bleached
root@svt-r1c1b01's password:
yourNewCertRequest
[root@dev-r1c4b09 ~]#
```

2. Browse to the directory of the Certificate Authority.
3. Sign the certificate using the following command: openssl ca –config openssl.cnf –notext –out <name of your signed request> -infiles <name of your personal certificate request>
4. After this command is run, a newly signed request will be created in the same directory.

Action Step 4: Importing the Signed Certificate into the WebSphere Environment

The newly created signed certificate now has to be transferred back to the filesystem of the WebSphere Deployment Manager. It will then be imported into the WebSphere Deployment Manager’s configuration.

1. Copy the signed certificate back to the file system of the WebSphere Deployment Manager and note the location of where you place it.

   [root@svt-r1c1b01 ca-bleached]# scp myNewCertSigned dev-r1c4b09:/myNewCertSigned
   [root@svt-r1c1b01 ca-bleached]#

2. On the WebSphere Administration Console, browse to: Security → SSL Certificate and key management → Key stores and certificates → CellDefaultTrustStore → Personal Certificates
3. Next you will select, “Receive from a certificate authority” to import the newly signed certificate.

4. Input the filename and location of which the signed certificate was copied and click OK.
5. The next menu will now show that the personal certificate (examplecert) has been added to the CellDefaultKeyStore.

6. To quickly verify the certificate was correctly recognized, browse back to the personal certificate request screen and verify that the original personal certificate request does not exist anymore. Browse to: Security → SSL Certificate and key management → Key stores and certificates → CellDefaultTrustStore → Personal certificate requests.
Action Step 5: Configuring the WebSphere Environment’s DefaultTrustStore and DefaultKeyStores

Now that we have both the public certificate from the certificate authority and our personal certificate request signed by the certificate authority; we must configure the proxy servers to use these certificates.

1. On the WebSphere Administration Console, browse to: Security → SSL Certificate and key management → SSL Configurations.

2. On the SSL configurations panel, changes will be made at both the cell and the nodes where the proxy servers are running.
3. Click on the NodeDefaultSSLSettings for the node the proxy server is installed on.

4. By default, the keystore name is set to “NodeDefaultKeyStore”. Change the keystore name to port to the keystore at the cell level.

5. After the “CellDefaultKeyStore” is selected, click on the “Get Certificate Aliases”. This will populate the default certificate aliases with the newly created certificates. Select the new certificate and click Apply and Save.
6. Repeat this process on each node there has a SIP Proxy Server associated.
7. Repeat this process at the cell level by modifying the “CellDefaultSSLSettings”.

Completion of Action Steps 1 through 5 will allow your SIP Proxy Server to accept TLS packets from the F5 BIG-IP LTM.

Using Converged (SIP/HTTP) Applications

Enable SNAT Automap
If the server nodes (proxies, in the case of this document) do not have the BIG-IP as their default route, an asymmetric routing situation will occur. In the case of UDP traffic for SIP, BIG-IP participates as a SIP proxy and inserts its own VIA header, creating the proper routing chain with the downstream proxy. In some cases, especially with TCP traffic, simply inserting the VIA header may not be enough. Therefore, if the proxy does not have the BIG-IP as its default route, a SNAT should be used to rewrite the address so that the proxy responds directly back to the BIG-IP in return traffic.

The benefit of using SNAT Automap is its ease of use. Simply turning on SNAT Automap in the Virtual server will use the Self-IP (as described in the paragraph above) and traffic will flow properly.

The downside of use SNAT Automap is that by using the one Self-IP address, there is a limit of approximately 64,000 ports open at one time. If you are anticipating more than 64,000 simultaneous TCP connections, then a SNAT pool should be used instead of SNAT auto map.

**SNAT POOLS**

If you decide that you want to use a SNAT pool as the way to specify translation addresses in your SNAT, you must first create the SNAT pool, specifying one or more translation addresses that you want to include in the SNAT pool. You create a SNAT pool using the Configuration utility.

**To create a SNAT pool**

1. On the Main tab of the navigation pane, expand Local Traffic, and click SNATs. The SNATs screen opens.
2. On the menu bar, click **SNAT Pool List**. This displays a list of existing SNAT pools.
3. In the upper-right corner of the screen, click **Create**. Note: If the Create button is unavailable, this indicates that your user role does not grant you permission to create a SNAT.
4. For the **Name** setting, type a unique name for the SNAT pool.
5. For the **Member List** setting, type an IP address.
6. Click **Add**.
7. Repeat steps 5 and 6 for each translation address that you want to add.
8. Click **Finished**.

After the SNAT pool has been created you apply it to the Virtual Server just as you would apply a SNAT auto map.