Direct Connect for Good Control/Good Proxy

Version 4.2

Contents

Revision history

Direct Connect

Date	Description
2017-09-19	Determining whether you should upgrade to BlackBerry UEM
2017-08-28	Version numbers updated for latest release; no content changes.
2017-07-18	Updated for latest release
2017-01-31	Version numbers updated for latest release; no content changes.
2016-12-19	Version numbers updated for latest release; no content changes.
2016-06-29	Version numbers updated for latest release; no content changes.
2016-03-10	Truncated revision history to reduce bulk.
2016-01-26	Added clarifying note to Configuring the F5 virtual server that unless a field and value is specifically called out, all values can be left at their defaults on the F5.
2016-01-15	Version numbers updated for latest release; no content changes.
2015-10-07	Added description of Enterprise CA certs with SSL-certificate-based client authentication
2015-10-12	Added new deployment configuration: Forward proxy with the F5appliance

Determining whether you should upgrade to BlackBerry UEM

If you require MDM or MAM capabilities, you must manage BlackBerry Dynamics apps using BlackBerry UEM. When you upgrade from Good Control to BlackBerry UEM, you not only get to use the great feature set that Good Control provides but you also get to take advantage of an enhanced feature set such as:

- Support for more policies for operating systems
- Better app management
- More container types
- Improved administration and provisioning
- Advanced connectivity and networking
- Expanded compliance and integrity checking
- Additional email, content, location, and certificate features
- Access to BlackBerry Web Services APIs

For information on how to use BlackBerry UEM to manage BlackBerry Dynamics apps, see the Getting started with BlackBerry UEM and BlackBerry Dynamics content.

For more information on the benefits of using BlackBerry UEM, see Benefits of upgrading from Good Control to BlackBerry UEM.

What's New in BlackBerry Dynamics Direct Connect

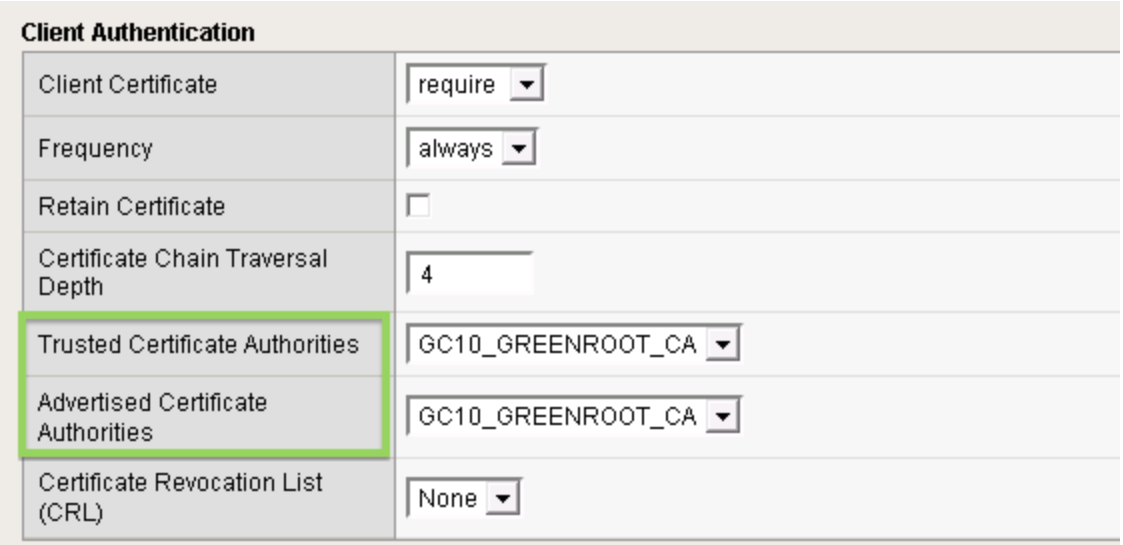
Enterprise CA certs with SSL-certificate-based client authentication

Previous versions of BlackBerry Dynamics supported mutual TLS authentication with a client certificate automatically issued by the BlackBerry Dynamics CA during provisioning. This functionality has now been extended to support enterprise-CA-issued TLS client auth certificates issued by the organization's own internal, enterprise CA, and synchronized to the BlackBerry Dynamics Runtime as a PKCS 12 file (with pfx or p12 filename extension).

The setup for SSL-certificate based client authentication with enterprise-CA-issued certs is similar to setup with the GC-issued certificate.

The certificate export/import onto the F5 or other appliance steps are the same as for the Good Control auto-installedcertificate created by the GC or GP during installation.

> **Important:** However, the appliance administrator must ensure that **Trusted Certificate Authorities** and **Advertised Certificate Authorities** are set on the appliance's client-side listener with the required details about the enterprise CA.



Correct configuration of **Advertised Certificate Authorities** is especially important, because the BlackBerry Dynamics Runtime uses this information in the TLS handshake to determine whether to send an enterprise-issued client certificate or send the default the BlackBerry Dynamics-issued client certificate.

BlackBerry Dynamics Direct Connect

BlackBerry Dynamics Direct Connect is a deployment option for the BlackBerry Dynamics Secure Mobility Platform. It delivers direct control over application data path, reduces round trip time (RTT), and enhances performance—all resulting in a superior user experience.

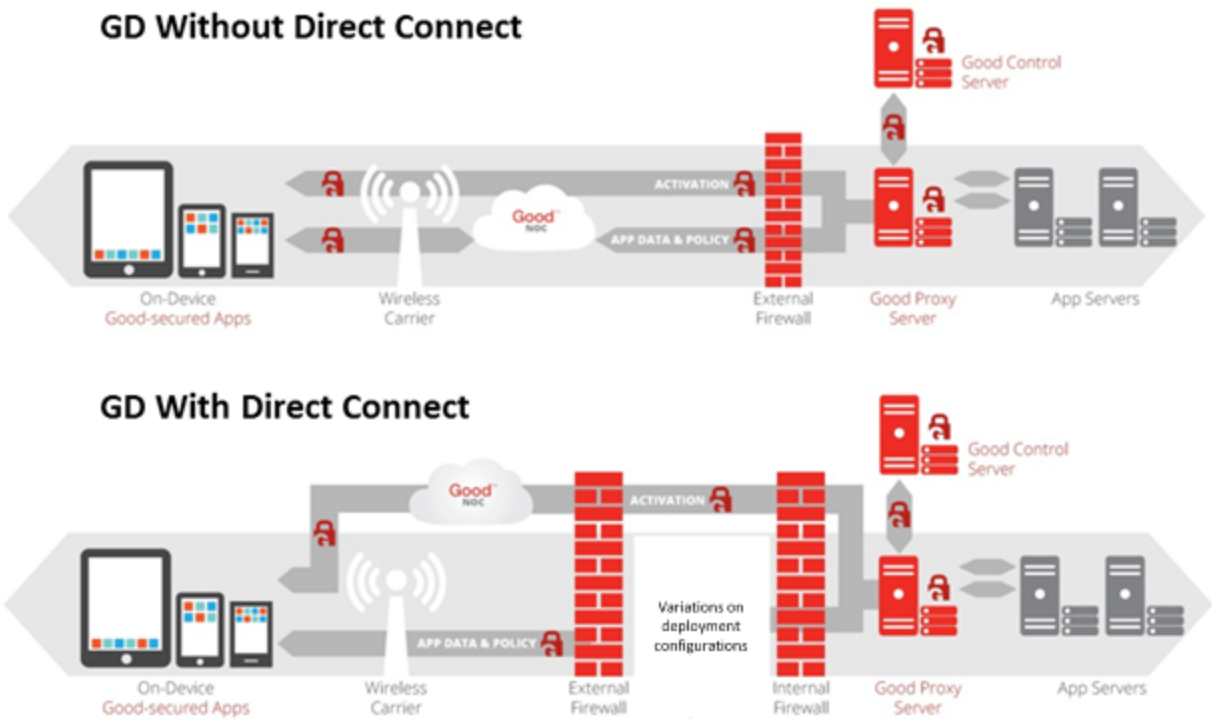
BlackBerry Dynamics Direct Connect has several benefits:

- Enhanced control because application data is always under corporate control, flowing directly to/from the corporate network, an important feature when your enterprise needs its sensitive data restricted to national and/or corporate boundaries.
- Improved network performance because BlackBerry Dynamics Direct Connect is a low-latency configuration allowing Good-secured applications to communicate directly with the Good Proxy server, thereby reducing data round trips to optimize bandwidth utilization for applications like HTTP video streaming.
- Better user experience because the reduced RTT lets applications refresh faster, contributing to a better overall user experience.

Before the introduction of BlackBerry Dynamics Direct Connect, the physical distance of users and organizations in the Eastern US, Europe and Asia from the BlackBerry Dynamics NOC servers located on the USA's West Coast potentially meant longer network RTT because of latency in connection establishment.

Direct Connect avoids this latency issue by allowing your enterprise BlackBerry Dynamics clients to establish direct connections with GP servers located behind the internal firewall, bypassing the BlackBerry Dynamics NOC servers to eliminate four long hops—from BlackBerry Dynamics client to BlackBerry Dynamics NOC, from BlackBerry Dynamics NOC to GP, then two hops back to the BlackBerry Dynamics client from the GP, thereby reducing RTT.

Below are high-level views of the BlackBerry Dynamics architecture, with and without Direct Connect. Direct Connect has four basic deployment models, which are detailed in Deployment configurations .



Depending on your organization's proximity to the BlackBerry Dynamics NOC, and assuming your BlackBerry Dynamics clients are situated closer to your GP servers than to the BlackBerry Dynamics NOC, the Direct Connect feature will likely improve the performance while reducing the latency of your BlackBerry Dynamics platform.

BlackBerry Dynamics Direct Connect does not eliminate the need for the BlackBerry Dynamics NOC, which is still required for application activation and authorization on client devices. Once provisioned and activated, Direct Connect affords you the flexibility to route application directly from your enterprise network to/from the application containers on the device, instead of having to go through the BlackBerry Dynamics NOC.

Direct Connect does not require any new BlackBerry components, although you can optionally use a standard commercial off the shelf HTTP proxy server to enable Direct Connect, rather than connecting to a GP server if so desired.

Direct Connect is not designed to provide better performance than a VPN. You should also not expect to see improvements when the BlackBerry Dynamics client is in close proximity to the BlackBerry Dynamics NOC.

About the BlackBerry Dynamics NOC and Direct Connect

Even with the Direct Connect configuration, it is important to know that the BlackBerry Dynamics Network Operation Center (NOC) is still a critical part of the architecture. It is always relied on for the following functions:

- Provisioning of applications on mobile devices.

- Notification of policy updates to active (currently open) BlackBerry Dynamics containers. For inactive containers, the policy update takes place the next time the container opens, but realtime notification requires a connection to the NOC.
- Applications that rely on the Secure Push Channel require connectivity to the NOC.

Other reliance on the BlackBerry Dynamics NOC with Direct Connect (or not) is pointed out in other sections of this document.

About BlackBerry Dynamics software version numbers

The cover of this document shows the base or major version number of the product, but not the full, exact version number (which includes "point releases"), which can change over time while the major version number remains the same. The document, however, is always current with the latest release.

If in doubt about the exact version number of a product, check the BlackBerry Developer Network for the latest release.

Relationship to Cloud GC: feature not applicable

The feature, service, server type, or software described in this guide is not available on Good Control Cloud because it is not applicable in a hosted environment.

Deployment configurations

Regardless of which DC deployment option is used, the following statements are always true.

- BlackBerry Dynamics NOC:
  - Provisioning of applications on mobile devices.
  - Notification of policy updates to active (currently open) BlackBerry Dynamics containers. For inactive containers, the policy update takes place the next time the container opens, but realtime notification requires a connection to the NOC.
  - Applications that rely on the Secure Push Channel require connectivity to the NOC.
- SSL/TLS : Communication between a client and the Good Proxy server is always secured over SSL/TLS.
- Access: By default all clients are denied access to the Good Proxy server.
- Good Control: An administrator retains all device management capabilities in Good Control.

Direct Connect does not change the security of the system. It simply provides an alternate way to deliver data from the client to the Good Proxy server. In the following section we will take a closer look at the various ways that DC can be deployed.

There are several key ways to deploy DC.

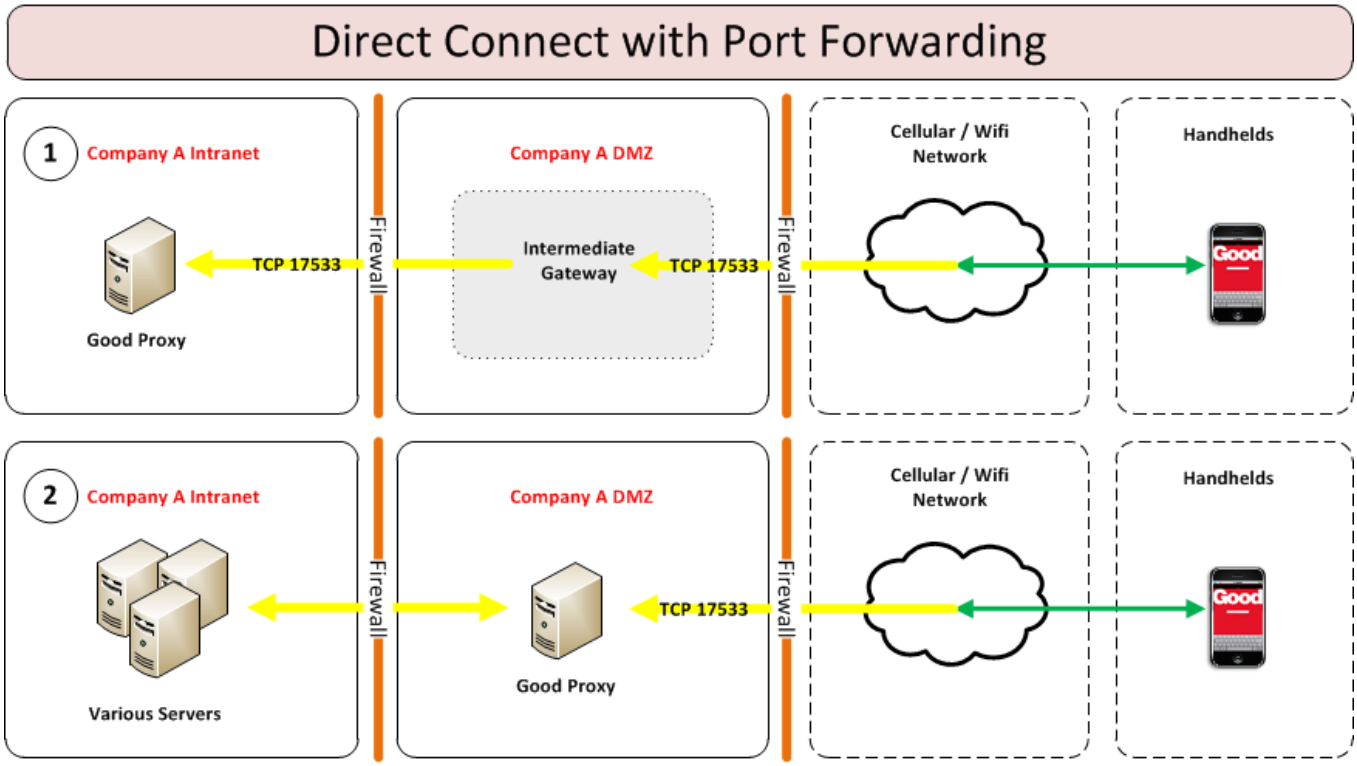
- Port forwarding
- fForward proxy without appliance
- Forward proxy with the F5appliance
- SSL bridging and a variation SSL-certificate-based client authentication including Enterprise CA certs with SSL-certificate-based client authentication

Port forwarding

This is the simplest deployment option for DC. In this approach we simply port forward all incoming client traffic to the Good Proxy server. There are two variations of this approach. The first variation is to port forward from the edge of the perimeter network directly into the corporate network where the Good Proxy resides. The Good Proxy server only requires one inbound port, TCP 17533. As long as the perimeter firewall is configured to only allow this port to the Good Proxy server then access is secured. As noted above, security policies are setup and managed in Good Control to allow access to the system.

The second variation of the port forwarding approach is to place the Good Proxy server in the corporate DMZ. The benefit of this approach is that you don't need to port forward directly from the edge of the perimeter network directly into the corporate network. Instead, you only need to port forward from the edge to the corporate DMZ network. However, additional ports will need to be open between the DMZ network and the corporate network in order to facilitate traffic between the Good Proxy and internal resources.

Both variations of the port forwarding approach are shown below.

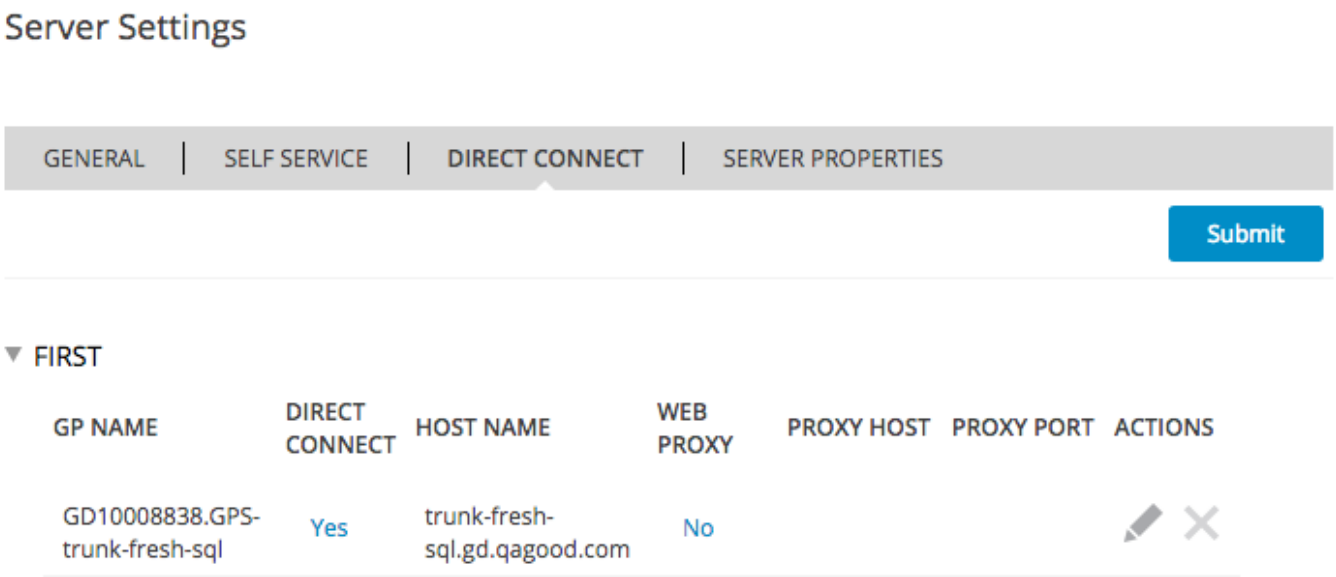


Port forwarding requirements

Regardless of which variation is used, a publicly routable DNS name is required for each Good Proxy server, for example, **`gp.mydomain.com`**. Depending on which method is used, the firewalls must be adjusted accordingly to forward TCP 17533. If you chose to place the Good Proxy server in the DMZ, then additional ports, including port

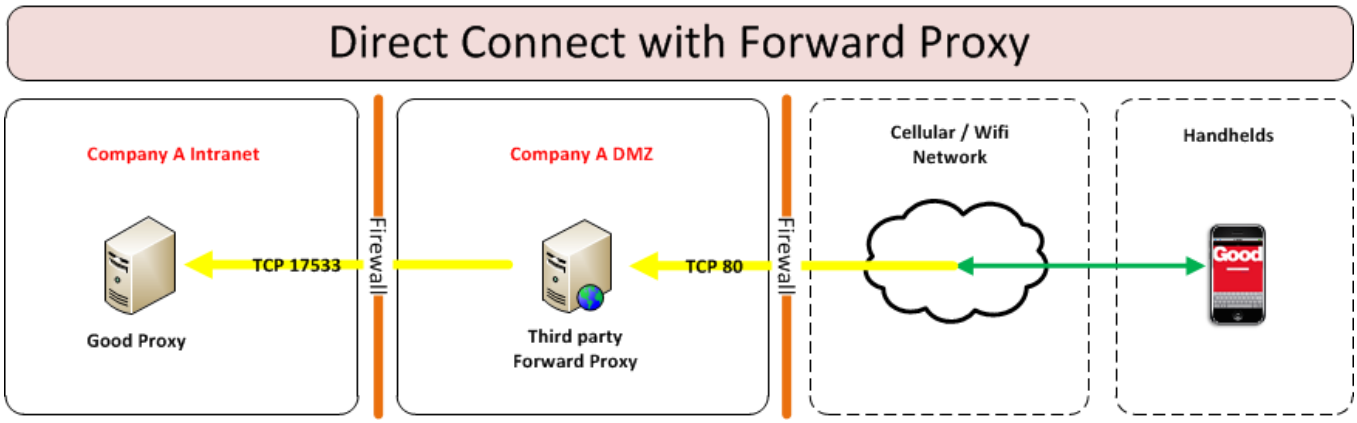
17433, need to be open between the DMZ and the corporate network. Other ports between the DMZ and corporate network vary depending on the resources that are required.

In Good Control, the Direct Connect configuration is accessible in the following menu on the left navigation area: **Servers -**> **Direct Connect tab**. The following is an example of the settings.



fForward proxy without appliance

In this DC deployment option, a forward proxy web server is used to proxy client requests to the Good Proxy server. The following diagram illustrates how this is deployed. For simplicity, only the vital components are shown.



The major benefits of this approach are:

- No need to port forward directly from the edge network to the internal corporate network.
- The forward proxy can load-balance incoming client traffic across multiple Good Proxy servers.

Forward proxy requirements

Direct Connect is agnostic with respect to forward proxying as long as the configuration meets the following requirements.

1. The forward proxy server must support the “HTTP CONNECT” method
2. The forward proxy must be able to communicate with the Good Proxy server via TCP port 17533
3. The forward proxy must be able to resolve the Good Proxy server's hostname.
4. An inbound port must be allowed to the Forward Proxy server. This port is arbitrary.
5. A publicly resolvable DNS hostname must be assigned to the Forward Proxy server.

As long as the above requirements are met any Forward Proxy servers can be used for direct connect.

In Good Control, the Direct Connect configuration is accessible in the following menu on the left navigation area: **Servers -**> **Direct Connect tab**. The following is an example of the settings.

Server Settings

GENERAL | SELF SERVICE | DIRECT CONNECT | SERVER PROPERTIES

Submit

▼ FIRST

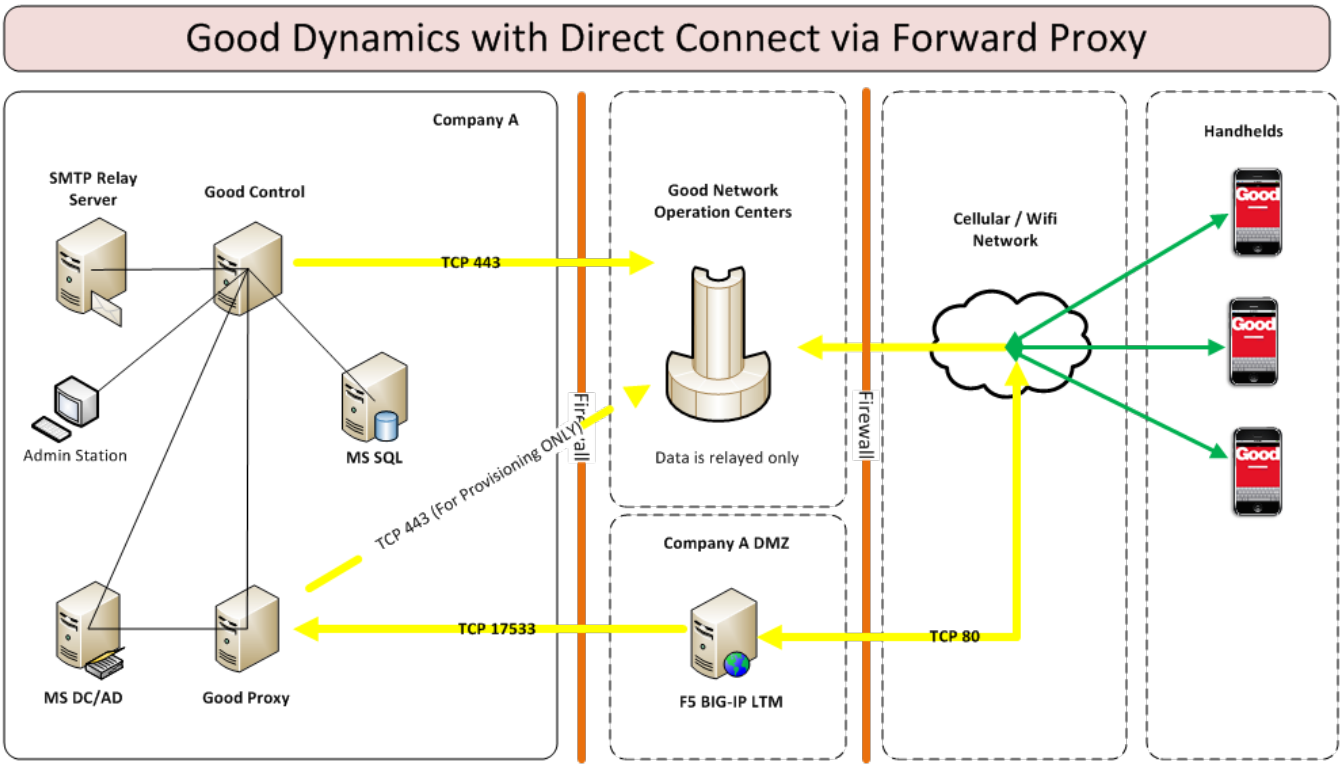
GP NAME	DIRECT CONNECT	HOST NAME	WEB PROXY	PROXY HOST	PROXY PORT	ACTIONS
GD10008838.GPS-trunk-fresh-sql	Yes	trunk-fresh-sql.gd.q	Yes	gp.mydomain.com	80	

Forward proxy with the F5appliance

In a forward proxy configuration, the F5 is configured as a forward proxy to facilitate the traffic between the client app and the Good Proxy server. Specifically, the F5 will act as a tunneling vehicle for the client app and the Good Proxy. The client app will initiate a "HTTP CONNECT" tunnel request to the F5. The request will contain the Good Proxy server that the client needs to connect with. If permitted by the F5, the request will be sent to the appropriate Good Proxy server. Once the tunnel is up, the client app will establish a SSL/TLS connection with the Good Proxy. Once again, all traffic between the client app and the Good Proxy server is facilitated via SSL/TLS.

The below diagram depicts the general architecture for this configuration. It is important to note that the incoming port from the client app to the F5 is arbitrary. The diagram shows 80; however, any port can be used as long as it is available. The port from the F5 to the Good Proxy must be 17533.

BlackBerry Dynamics DC with Forward Proxy



F5 BIG-IP LTM configuration

General configuration of the F5 BIG-IP LTM server is outside the scope of this document. Instead, this section will cover specific configuration as it pertains to BlackBerry Dynamics Direct Connect.

Note: the instructions listed below are based on version 11.5.1 build 0.4.110. Screen shots and instructions may vary on different versions.

Configuring forward proxy

By default the F5 server does not have a setting for "Forward Proxy". Instead, it is up to the administrator to create the necessary configurations to implement a forward proxy. The most common way to do this is to create an iRule that emulates a forward proxy. An example can be found in F5's DevCentral.

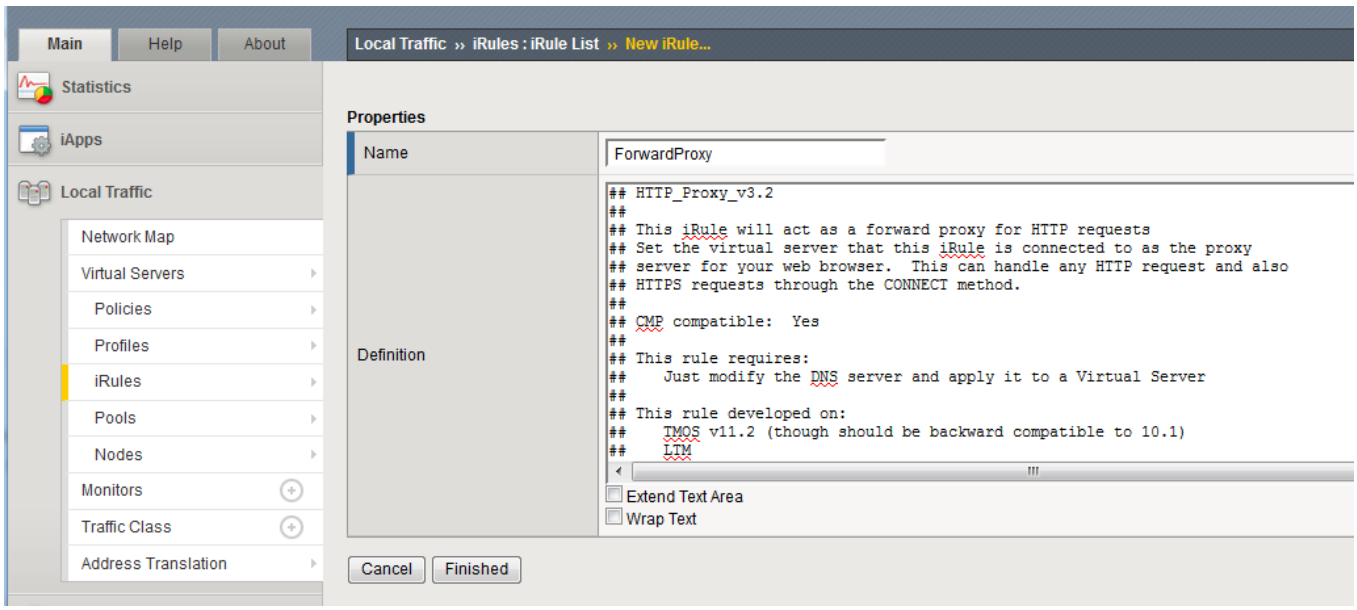
https://devcentral.f5.com/wiki/irules.HTTP-Forward-Proxy-v3-2.ashx

We will use this example for the rest of the configuration; however, please make sure you understand how this iRule works before applying it on your system.

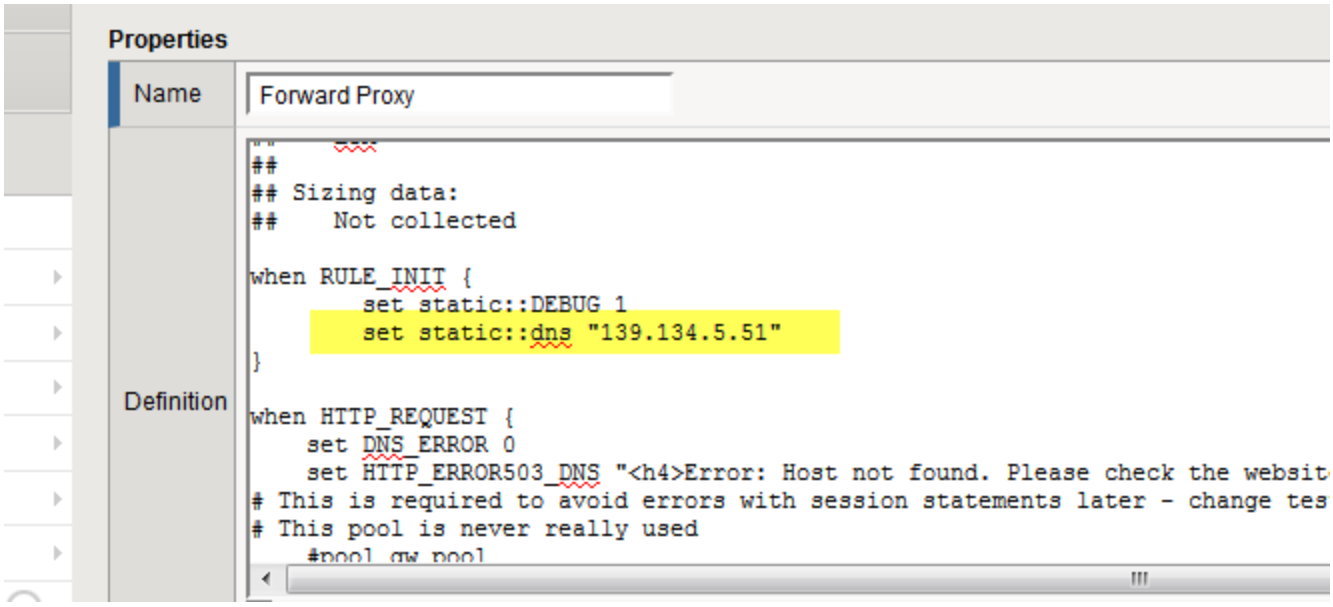
procedure – create irule:

1. The first thing that needs to be done is to copy the script. Hoover your mouse over the right hand corner of the script. Three options should appear. Use the first one to copy the script (see below).

2. Login to the F5 console. From the **Main tab -> Local Traffic -> iRules**
3. Click **Create** and then fill in the name and paste in the script.



Before clicking **Finish**, update the DNS server value to reflect your DNS server.



4. Done

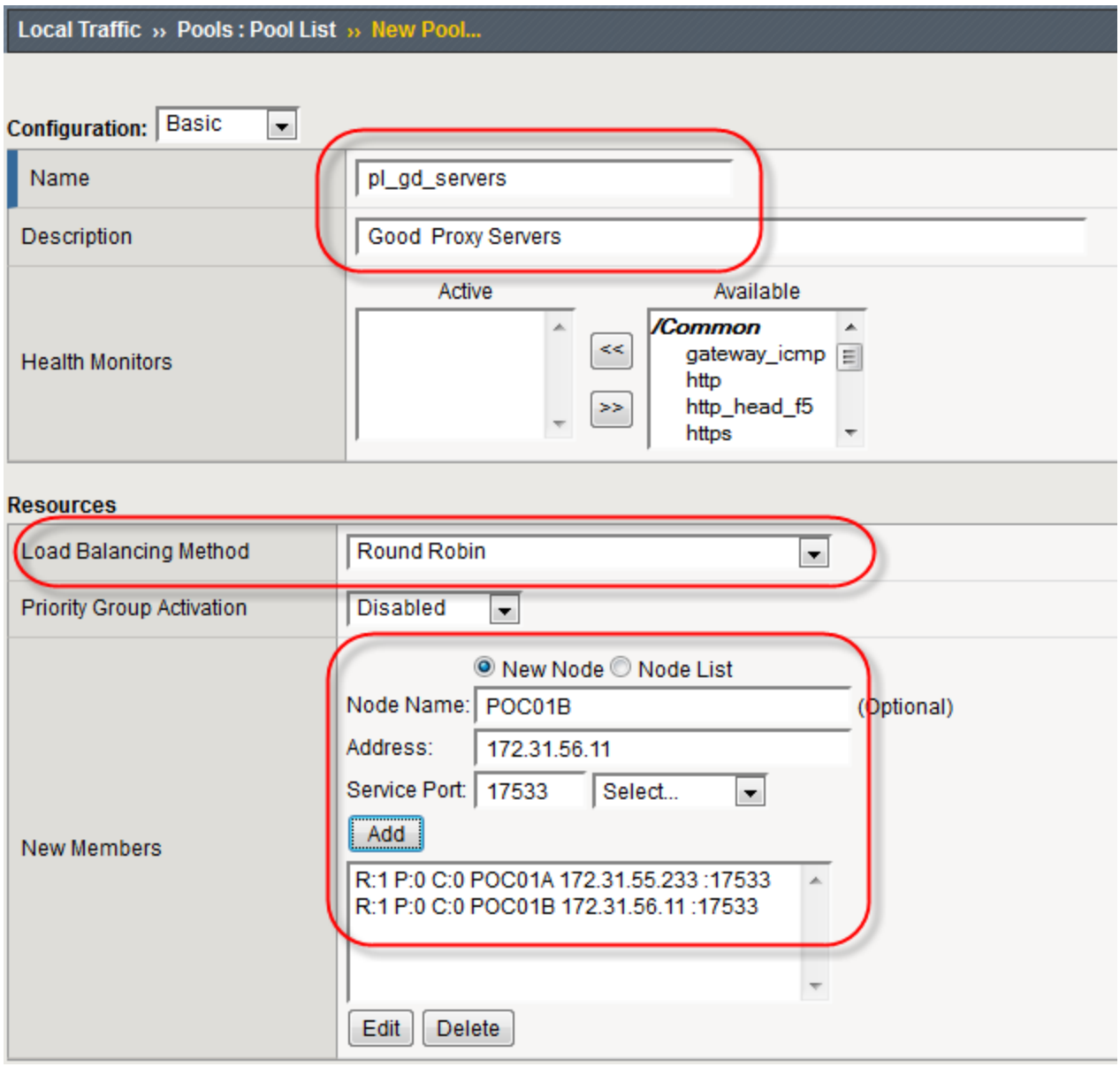
Procedure – create virtual server

1. Create Virtual Server :From the Main tab -> Local Traffic -> Virtual Servers
2. Click Create and note the following settings

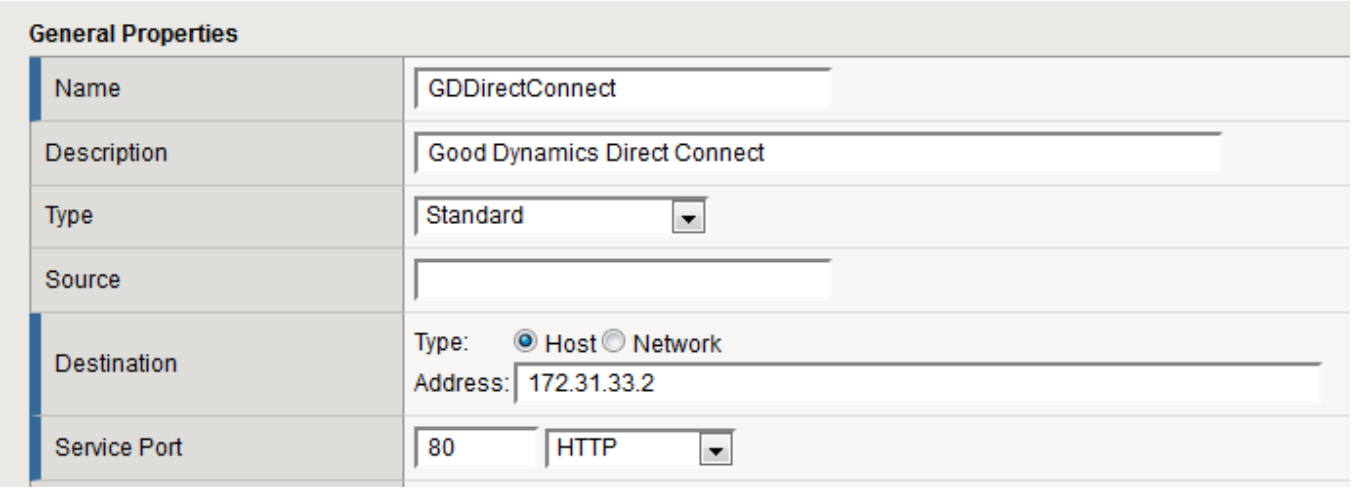
a. Name –this is arbitrary

b. Destination – this should be a publicly accessible address or an internal address that is NATTed to a publicly accessible address. The type should be host.

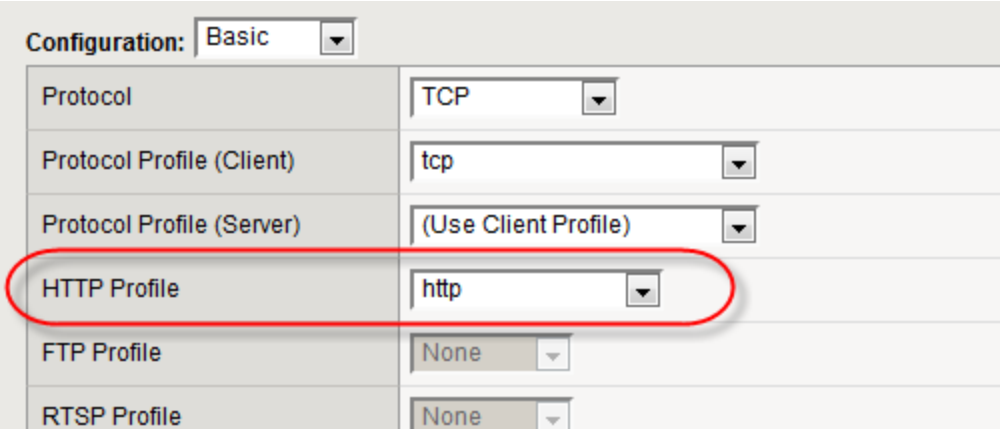
c. Service Port – this is arbitrary as long as it is available.



d. HTTP Profile – should be HTTP (use the default one)



e. Source Address Translation – set to Auto Map



f. iRules – select the ForwardProxy iRule that was created earlier

g. Click **Finish**

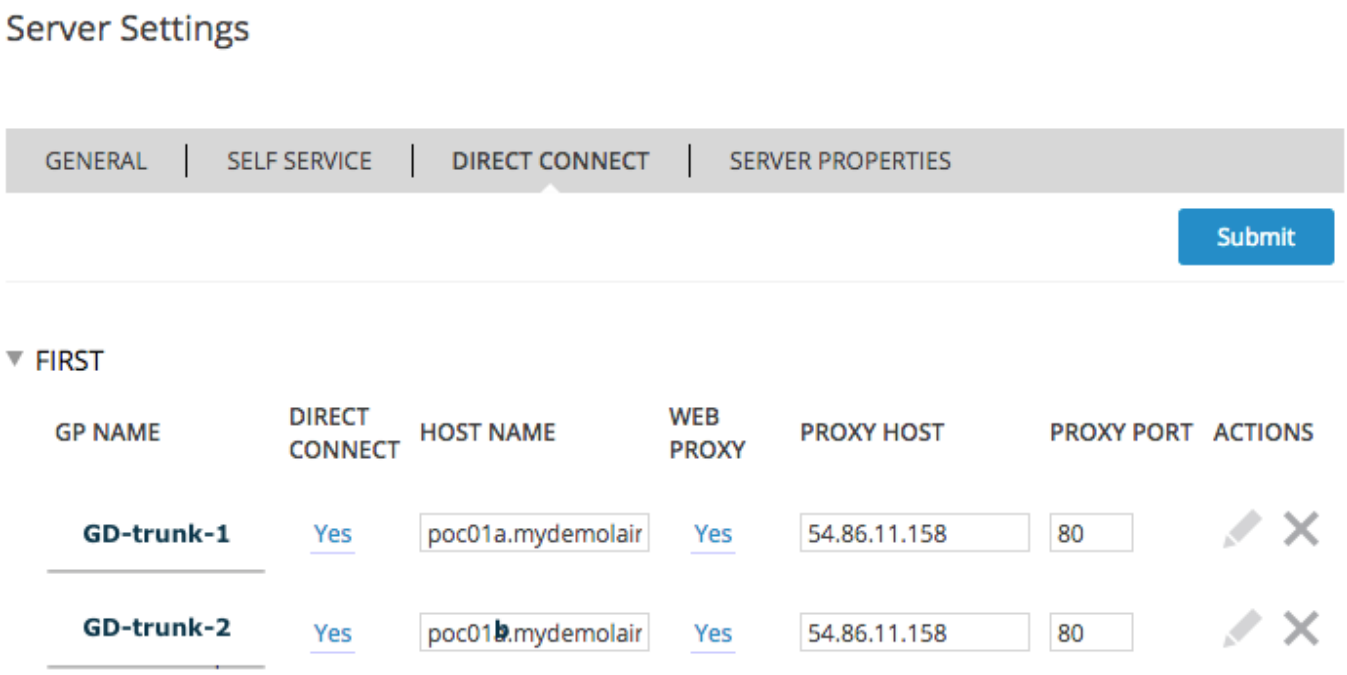
8. Done

GC Direct Connect for forward proxy

Login to the Good Control web portal to complete the following procedures:

1. Click **Settings -> Direct Connect**
2. For each Good Proxy server that will participate in Direct Connect, update the following fields

a. Direct Connect: Yes
b. Host name: this has to be a DNS that resolves to the respective GP server. This value cannot be an IP address
c. Web Proxy: Yes
d. Proxy Host: this value needs to the publicly accessible IP address or DNS name of the F5.
e. Proxy Port: this is the external port that the F5 will listen on.
f. The settings should look something like this:

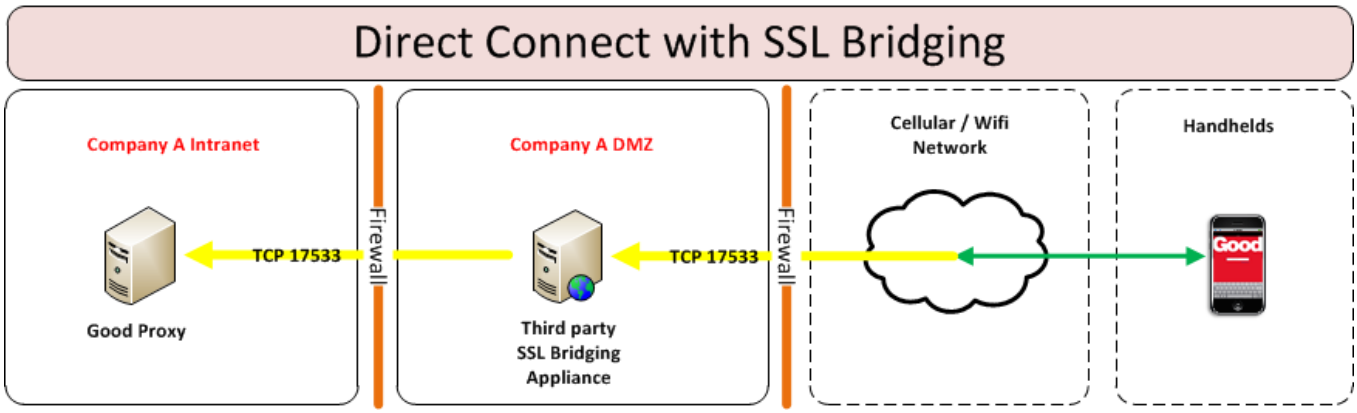


Click Submit to save the changes.

3. Done

SSL bridging

This deployment option is the most complex. This option involves using a third-party appliance to terminate the SSL/TLS connection from both the client and the Good Proxy server. The third-party appliance then bridges the two connections. The architecture is as follow:



The benefit of this approach is that the third-party appliance may be able to do additional filtering of the incoming traffic before sending it to the Good Proxy server. Load balancing of the incoming client traffic can also be achieved. However, these functions are highly dependent on the third-party appliance that is used. Configuration of these features is beyond the scope of Direct Connect. Consult your appliance manufacturer's documentation.

SSL bridging requirements

Direct connect is agnostic to the third-party SSL bridging appliance that is used as long as it meets the following requirements:

1. The bridging appliance must be able support the following ciphers

a. TLS_RSA_WITH_AES_256_CBC_SHA256 OR

b. TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384

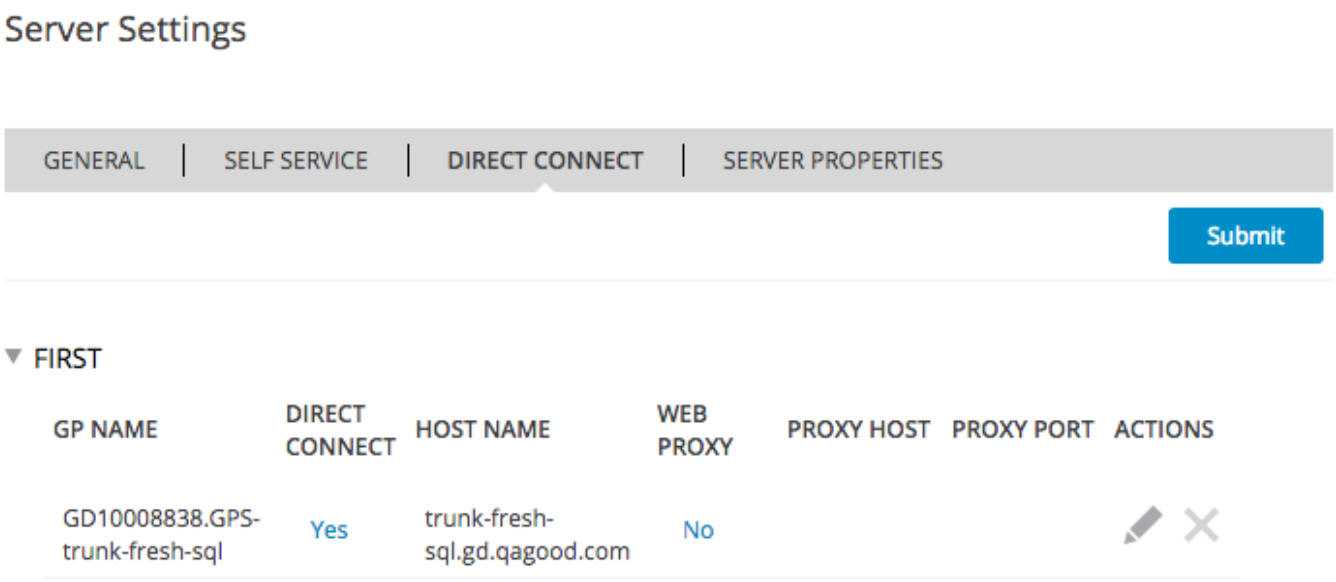
2. Inbound TCP 17533 must be opened to the appliance.

3. Outbound TCP 17533 must be open from the appliance to the Good Proxy server.

4. A publicly resolvable DNS hostname must be assigned to the appliance for the purpose of DC.

As long as the above requirements are met any third-party SSL bridging appliance can be used for DC. An example of how to configure a F5 BIG-IP LTM appliance for SSL bridging for DC is in Direct Connect with SSL termination at reverse proxy .

In Good Control, the Direct Connect configuration is accessible in the following menu on the left navigation area: **Servers -> Settings -> Direct Connect tab**. The following is an example of the settings.



Note: This Good Control configuration is exactly the same as the Port Forwarding deployment model.

SSL-certificate-based client authentication

This deployment option is a variation on SSL bridging . In SSL bridging, an SSL termination device (or "appliance", such as Netscaler or F5) terminates the incoming direct connect connections and acts as a bridge between users' mobile devices and the GP. The SSL listener or connection point on the bridge contains the same public and private key that resides on the GP server the connections are destined for. In this configuration, connections from the mobile devices are terminated on the appliance, and a new secure channel is created from the appliance back to the GP. This allows true termination at the edge for incoming connections.

During the provisioning process of each BlackBerry Dynamics secured application, a certificate signing request (CSR) is generated by the application and signed by the GDCA (BlackBerry Dynamics Certificate Authority). After being signed, this Inter-Container Communication certificate (or “ICC cert”) is sent to the application, where it is secured inside the secure container of the application. The GDCA that signs the ICC certificate is the root CA of a client’s specific BlackBerry Dynamics environment, and is the same root CA that signs the certificates on the GP servers and also the certificate exported to the listener of the SSL termination device where the SSL connection for Direct Connect terminates in this configuration.

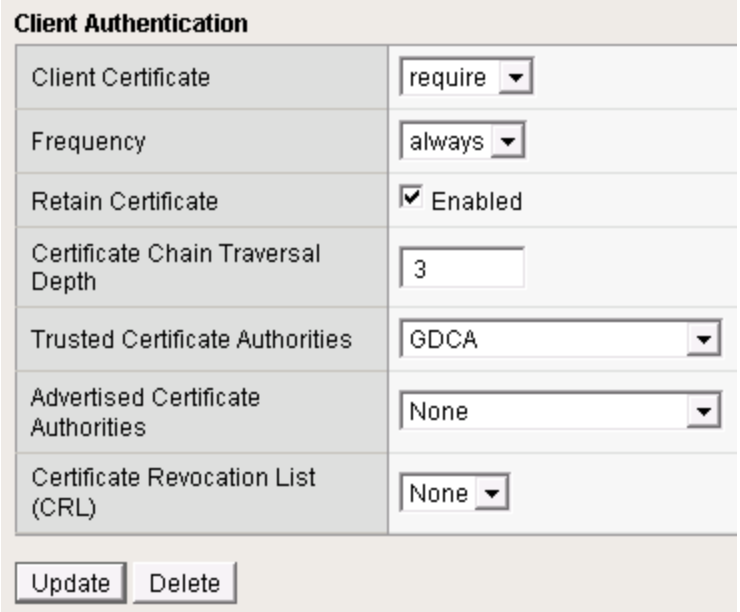
Every application has an ICC certificate. This certificate is specific to the application, its specific BlackBerry Dynamics environment used for key generation, and device on which the application is installed. The certificate is used for Good's patented Shared Services Framework (also know as AppKinetics) for inter-application transfer of files/data, such as “open-in” functionality. The SSL termination device’s listener (endpoint) is responsible for issuing the challenge for presentation of a client certificate during the Direct Connect TLS 1.2 channel establishment from the application to the appliance. *Therefore, the appliance must have a copy of the GDCA certificate.* In the negotiation phase, once challenged, the application presents the ICC certificate to the appliance's listener, which then validates the ICC certificate against the GDCA certificate authority. The application also validates that the certificate presented by the appliance during the negotiation is signed by the GDCA, because the application has a copy of this root certificate in its secure container, as well. After both certificates are validated, the connection is considered authenticated and the TLS channel is successfully established. If the ICC cert is signed by any certificate authority other than the same GDCA that is configured on the appliance’s listener, the authentication fails and no TLS channel can be established.

Setup Requirements

1. Make sure that your communications appliance supports SSL-certificate-based client authentication.
2. Set up Direct Connect with the SSL bridging deployment configuration.
3. Configure the SSL listener on your appliance to require client certificate authentication. The device must also be configured to validate the presented client certificates against the GDCA root certificate, which must be exported from the GC and imported into your appliance. The exact steps on establishing this appliance requirement vary from one vendor to another. Consult your appliance vendor's documentation.

   Below is an example of the relevant settings on the F5, which come at the bottom of the Client-SSL Profile section.

Note the name of the Certificate Authority: GDCA.



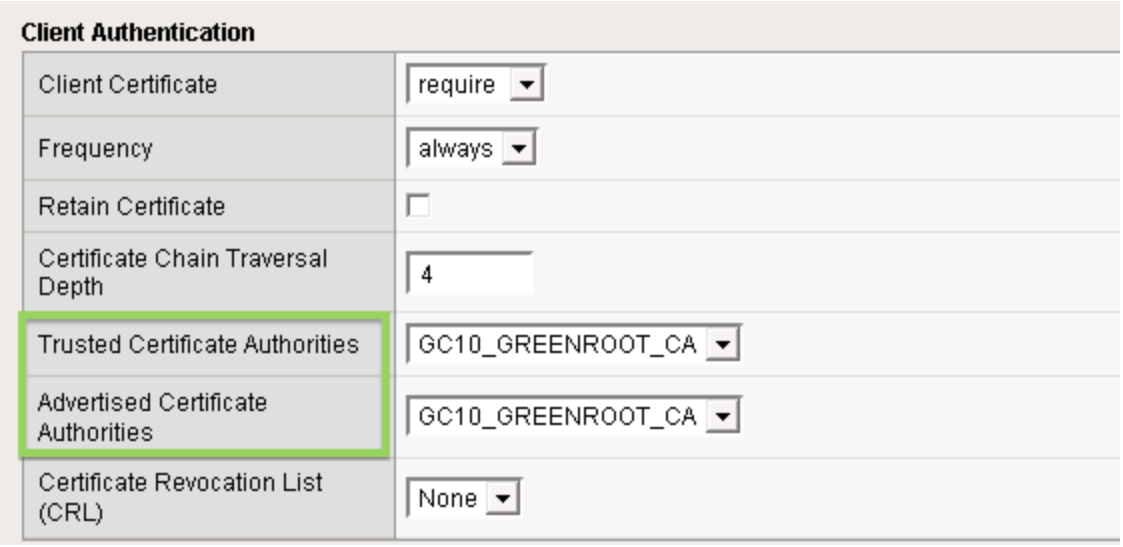
Enterprise CA certs with SSL-certificate-based client authentication

Previous versions of BlackBerry Dynamics supported mutual TLS authentication with a client certificate automatically issued by the BlackBerry Dynamics CA during provisioning. This functionality has now been extended to support enterprise-CA-issued TLS client auth certificates issued by the organization's own internal, enterprise CA, and synchronized to the BlackBerry Dynamics Runtime as a PKCS 12 file (with pfx or p12 filename extension).

The setup for SSL-certificate based client authentication with enterprise-CA-issued certs is similar to setup with the GC-issued certificate.

The certificate export/import onto the F5 or other appliance steps are the same as for the Good Control auto-installedcertificate created by the GC or GP during installation.

Important: However, the appliance administrator must ensure that **Trusted Certificate Authorities** and **Advertised Certificate Authorities** are set on the appliance's client-side listener with the required details about the enterprise CA.



Correct configuration of **Advertised Certificate Authorities** is especially important, because the BlackBerry Dynamics Runtime uses this information in the TLS handshake to determine whether to send an enterprise-issued client certificate or send the default the BlackBerry Dynamics-issued client certificate.

Testing BlackBerry Dynamics Direct Connect

To test and verify your Direct Connect connectivity, we recommend using a custom application built with the latest BlackBerry Dynamics SDK for iOS or for Android, or you can verify using one of the BlackBerry Dynamics sample applications included in the downloaded SDK bundle; for instance, the RSSFeed sample app.

Additional considerations

The Good Proxy "external" address only needs to be reachable from the Internet if no HTTP proxy is used. If a HTTP proxy is configured, then only the HTTP proxy address needs to be Internet accessible. The GP "external" address, in this case, would only need to be accessible from the HTTP proxy.

Direct Connect is configured on an individual Good Proxy basis. This means you won't be able to configure Direct Connect at the cluster level. It is therefore recommended as a best practice to make sure all GPs in a cluster are configured for Direct Connect, since GPs in a cluster are chosen at random. Consequently, if some GPs in the cluster are DC while others are not, you will continue to have some connections going through the BlackBerry Dynamics NOC arbitrarily.

Frequently asked questions

Included here are some of the most commonly asked questions regarding the BlackBerry Dynamics Direct Connect feature.

Q. Are there any special requirements when na HTTP proxy is used for implementing BlackBerry Dynamics Direct Connect?

A. A customer can use a standard off the shelf (OTS) HTTP proxy server as long as it supports the HTTP connect command and does not require separate authentication.

Q. Why would I choose to use the optional HTTP proxy?

A. You should choose the configuration that makes the most sense for your organization and environment. For instance, you may opt to use a HTTP proxy in the DMZ to reduce the maintenance cost of adjusting the internal firewall to allow connections between the Good Proxy and newly white-listed app servers. Even so, in cases where all connections to on-premise servers go through a proxy on campus, using Good Proxy may be the more suitable installation option.

Q. Can a reverse proxy be used when implementing Direct Connect?

A. Yes. See the details for configuring Direct Connect with a reverse proxy in this document.

Q. If there is no authentication at the HTTP proxy server level, is the Direct Connect as secure as the standard configuration, which relays data through the BlackBerry Dynamics NOC?

A. Generally when a HTTP proxy is put in the DMZ, authentication is required because the proxy is the access point to anything within the enterprise. This stricture is accommodated by configuring the DMZ-based HTTP proxy to only allow a path to the behind-the-firewall Good Proxy server using the specified port and address. Anything identified that is not explicitly configured on the DMZ-based HTTP proxy will not be allowed to go through the enterprise firewall, thereby restricting external access to the GP server, which performs the authentication, then allows the perimeter infrastructure to do its business and take care things like DPI, DOS detection/prevention, and so forth.

Q. Is BlackBerry Dynamics Direct Connect supported in HA/DR scenario?

A. Not only can BlackBerry Dynamics Direct Connect be enabled in a HA/DR scenario, it is a recommended configuration so you can take advantage of your designated fail-over path. You can configure one DMZ-based HTTP proxy server for multiple Good Proxy instances or distinct DMZ-based HTTP proxy servers for each Good Proxy server.

Essentially, you could set up the primary cluster of Good Proxy servers to use the BlackBerry Dynamics Direct Connect feature and point those Good Proxy servers to a single DMZ-based HTTP proxy server address. You can then designate a secondary cluster of GP servers to use another DMZ-based HTTP proxy server. Or, you can choose not to enable Direct Connect for that secondary cluster of GP servers.

Q. If the HTTP proxy in the DMZ fails and HA/DR has not been used, will clients fail over to the BlackBerry Dynamics NOC? Can this failover be turned off for regulatory reasons??

A. With Direct Connect, connectivity to App Servers will adhere strictly to configurations set in the Good Control server. If you don't provide a non-Direct Connect path to an App Server, the client app will never connect through the NOC. However, for connectivity to BlackBerry Dynamics servers, if only Direct Connect paths are configured and the BlackBerry Dynamics Library is unable to reach any BlackBerry Dynamics server via these Direct Connect paths, then it will fail over to connecting through the NOC. This is done to ensure that any policy updates, server configurations/addresses, and proxy configurations/addresses remain current.

Q. Can settings for the DMZ-based HTTP proxy be updated? If so, how quickly can these setting be received by a client app?

A. You can update the addressing information for the DMZ-based HTTP proxy at any time from the Good Control management console. Receipt of that new addressing information by the client app is immediate if the app connected to the network. If not connected, then the new addressing information is received immediately upon the next connection.

Q. Does the client app always need to connect through the BlackBerry Dynamics NOC when there is a connection failure to the DMZ-based HTTP proxy?

A. The complete BlackBerry Dynamics Direct Connect addressing information is sent to the client at activation and again whenever this information is changed. If more than one HTTP proxy server is in use in an HA/DR scenario, the client does not need to reconnect to the BlackBerry Dynamics NOC after a connection failure in order to get the address of additional HTTP proxy servers, since it already has all of this information.

Q. If I implement Direct Connect, do I need to restart the BlackBerry Dynamics servers?

A. You do not need to restart the BlackBerry Dynamics servers if you change the BlackBerry Dynamics Direct Connect setting. Changes are transparent to the end user.

Q. Do the BlackBerry Dynamics servers monitor the health of the proxies used for BlackBerry Dynamics Direct Connect?

A. The BlackBerry Dynamics servers do not monitor the health of any HTTP web proxies used for Direct Connect. You are therefore encouraged to configure multiple DMZ-based proxies, as well as to monitor proxy health using other off the shelf network monitoring tools.

Q. How is app data secured with Direct Connect?

A. Not only is traffic end-to-end encrypted but in the Direct Connect case, where SSL is used to secure the link between the client and the Direct Connect relay (or load balancer) the client only accepts certificates that come from a CA that is under the control of the customer and so is not subject to attacks through the coercion of a commercial CA.

Q. For SSL bridging or proxying, how can I change/add to the SSL ciphers that can be allowed?

A. By default, SSL communications between the GC and GP servers over port 443 for the Direct Connect configuration uses the following ciphers:

- TLS_RSA_WITH_AES_256_CBC_SHA256 OR
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384

If you need to add more ciphers, after installation, edit the GP server's configuration file **`c:\good\gps.properties`** and add the names of the ciphers to the **`gps.directconnect.supported.ciphers`** key.

One reason you might need to add more ciphers is if you have your own proxy server between your client devices and the GP server configured for Direct Connect. This middle proxy is the one that determines which SSL ciphers to use. You need to ensure that the GP server ciphers correspond to those required by your own proxy.

Q: Some of our BlackBerry Dynamics applications were compiled with older versions of the BlackBerry Dynamics SDKs that do not support Direct Connect. Will they have issues connecting if we move to Direct Connect?

A: Yes. This will cause an issue and older apps will not be able to connect. One way to mitigate this problem is to configure Direct Connect at the application level, so some applications communicate via Direct Connect(those applications those that support Direct Connect), and other applications connect via BlackBerry Dynamics NOC. However, to support this configuration, you will need multiple GP clusters.

Direct Connect with SSL termination at reverse proxy

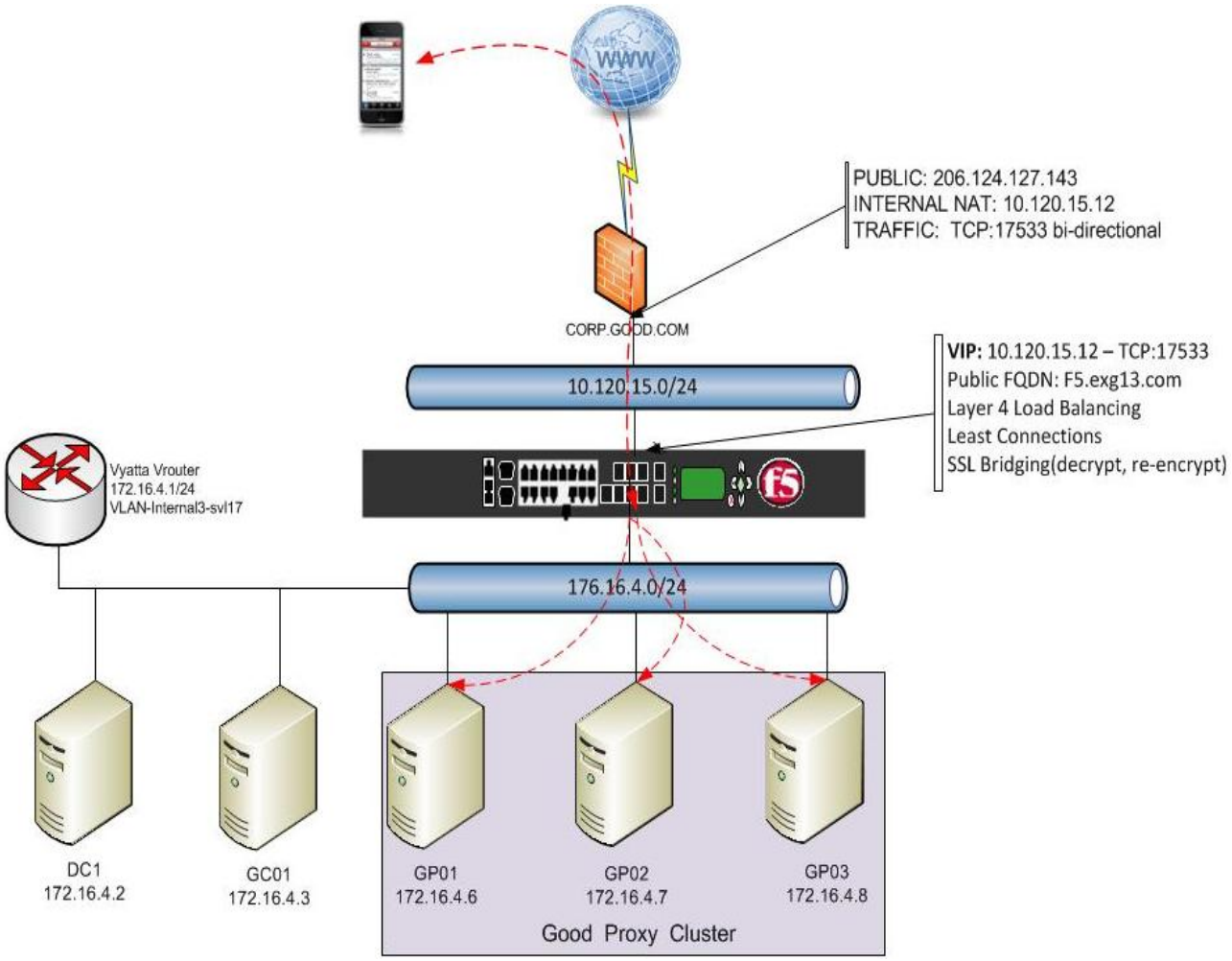
BlackBerry Dynamics Direct Connect is currently supported in two main deployment models:

- Direct Connect with no Web Proxy
- Direct Connect with a Web Proxy

While both of these methods are supported, many enterprises prefer to use an edge network device that will terminate the SSL connection from the device as it ingresses into the corporate network at internet edge. Upon connection to the edge device, the application can establish connections to any of the Good Proxy servers defined in the cluster specified for Direct Connect.

The following diagram showsthe network architecture, subnets, location of reverse proxy(F5), and traffic flow as applicable to Direct Connect.

This architecture allows a single publically exposed IP address to accept connections for all servers within a GP cluster.



Creating the key pair for external listener on F%

The BlackBerry Dynamics platform uses a proprietary method for signing, securing, and distribution of certificates used in the communication process between GC and GP servers, along with communications between BlackBerry Dynamics secured applications and GP servers.

For this initial testing, it is required to use some open source SSL key tools to create the necessary key pair required for utilizing the F5 as a reverse proxy.

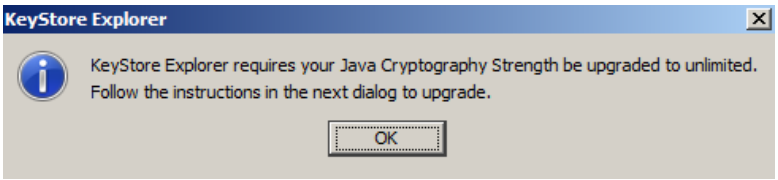
Installing the key store explorer

Some recommendations before you begin:

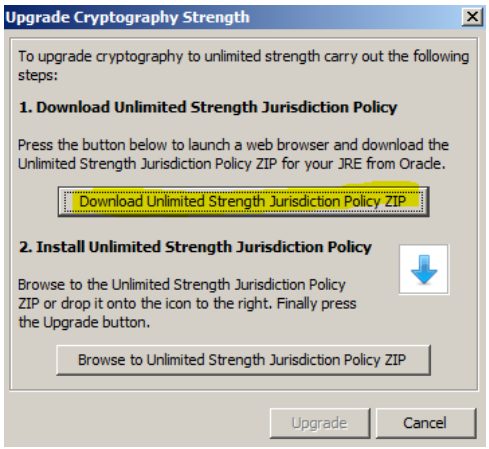
- Make a backup of the GC server's *`installation_directory`***`\jre\lib\security\cacerts`** file.
- BlackBerry recommends that for testing you install the key store explorer on a server external to your intranet and not on the GC server. GC has its own copy of the Java Runtime Engine (JRE) that is probably different from that required by the key store explorer.
- From this separate server, make sure you have read/write access to the GC server's file system.
- If you must install the key store explorer on the GC, install Java for it in a directory that is separate from the GC Java directory. If the values of the GC's JAVA_HOME and JRE_HOME environment variables have to be modified, after testing make sure to reset the variables to their original values.

Steps

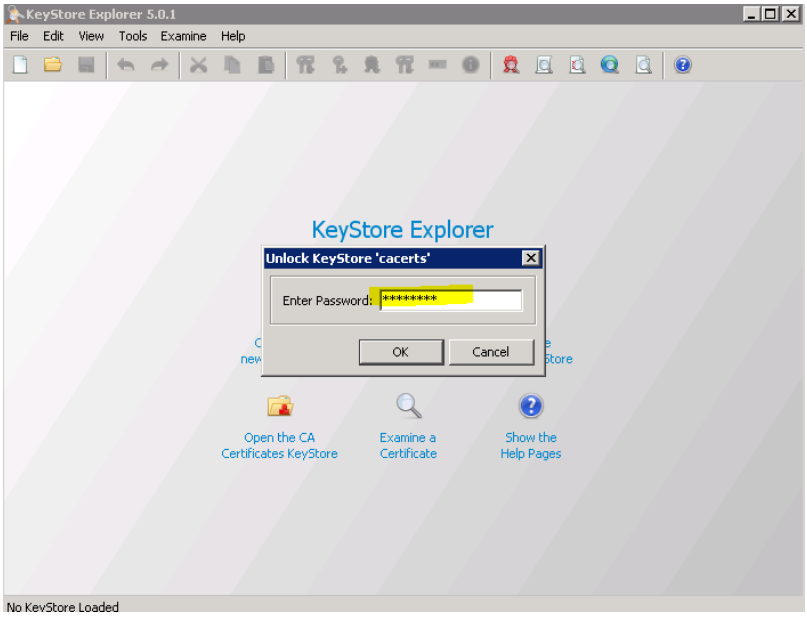
1. Download the required version for your operating system onto your separate external-to-the-intranet server or the Good Control server:
   http://keystore-explorer.sourceforge.net/downloads.php
2. After launching the application you will be required to install Java if it is not already installed. Clicking “OK” if Java is not found will direct you to the appropriate download site.
3. Upon re-launch of the Keystore-Explorer application you will be prompted to upgrade your Java Cryptography Strength to unlimited.



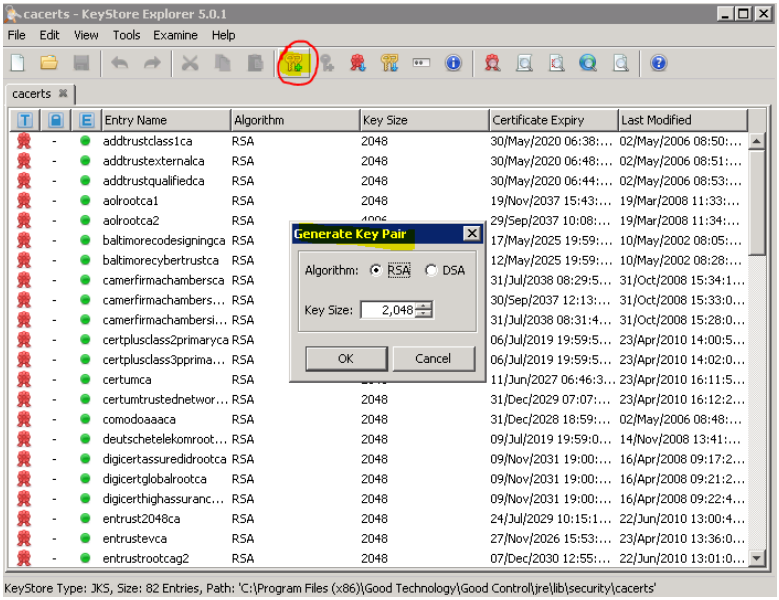
4. Pressing OK will direct you to appropriate site to download the appropriate zip file.



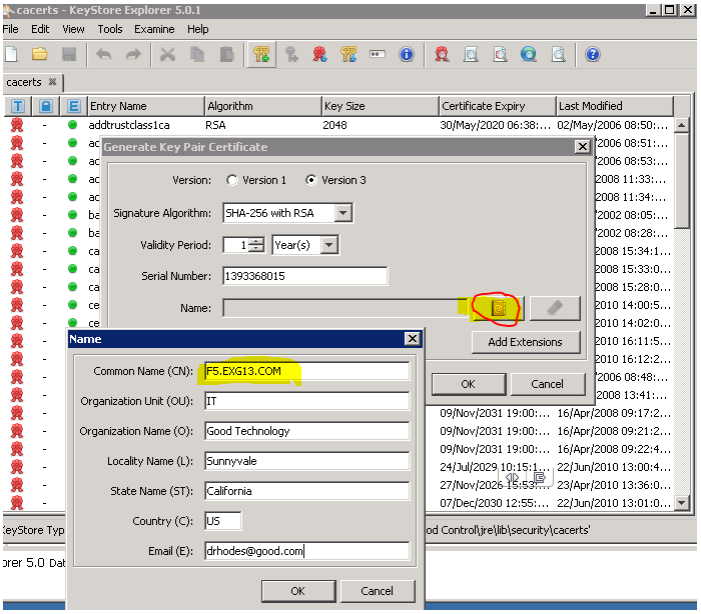
5. Save the zip file in a known location on your local PC and then browse to the saved zip file for Keystore-Explorer to import required files and click “Upgrade”
6. In the Keystore tool, select “Open an Existing Keystore” and browse to the following installation directory on your Good Control server: *installation_directory*`\jre\lib\security`.
7. Select the file `cacerts` to open in the Keystore Explorer. When prompted for password, the default password is `changeit` all lowercase.



8. Generate a Key Pair by clicking on the icon as shown below and then selection OK, accepting the auto populated values.



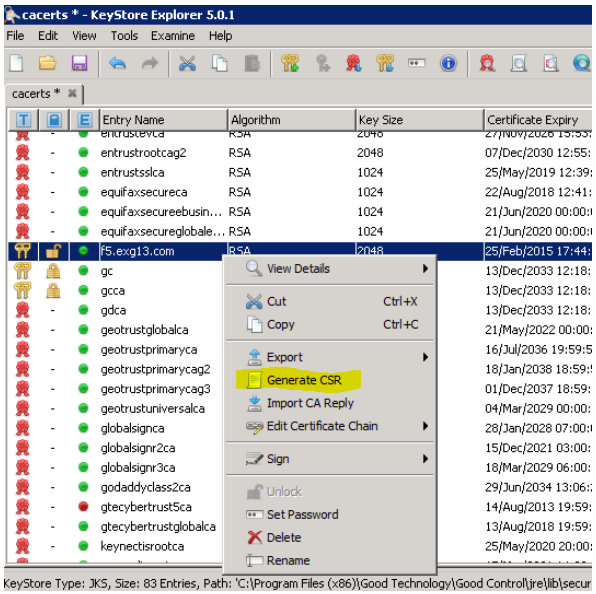
9. After the key pair is generated, you will be prompted with the screen below, Click on the highlighted section to change the certificate attributes to match your environment. The “Common Name (CN)” field must be populated with the FQDN of the F5 listener. This is the name resolvable from the public Internet for which this certificate will be valid. In this example F5.EXG13.COM is the name of the listener.



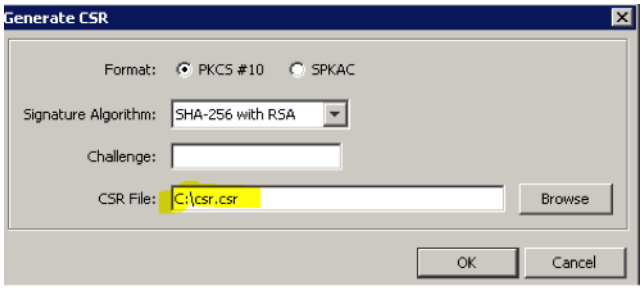
10. Click OK, then OK to accept the default alias which should be the CN you populated in above step, and input the password and confirm password.

> **Important:** You must use the password **`changeit`** because the web server associated with the GC expects this password.

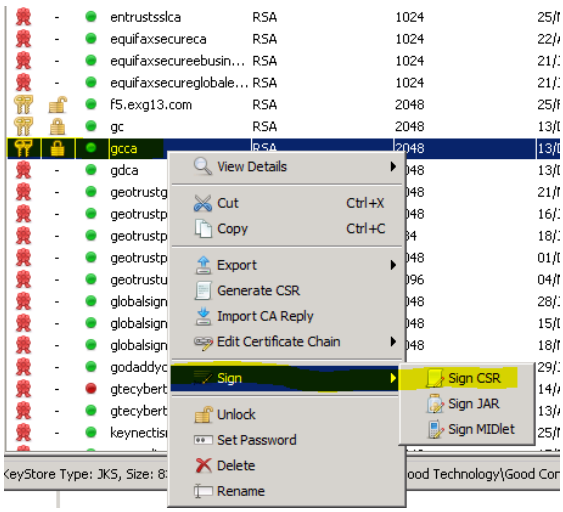
11. Next generate a CSR from this key pair by right-clicking and selecting **Generate CSR**.



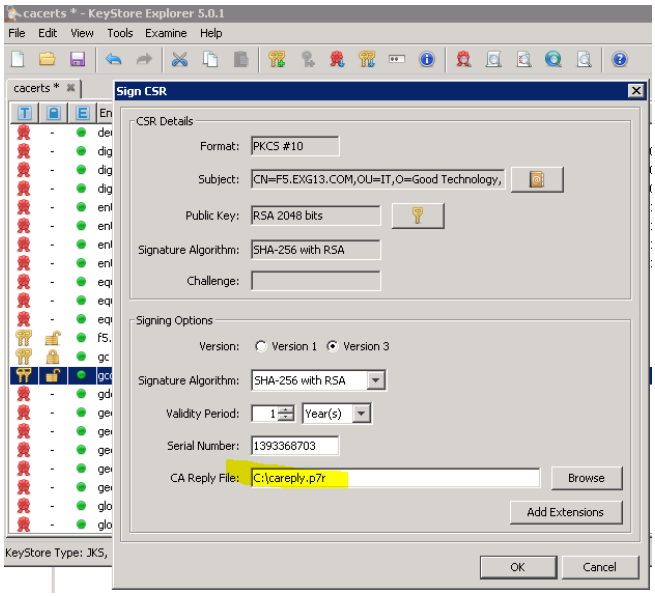
12. Leave the default Format and signature Algorithm and enter a location and name for the CSR.



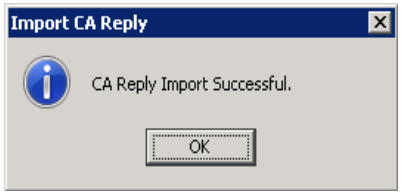
13. Right-click the “gcca” key and choose to sign the csr file you just created. You are prompted for the same password to unlock the gcca key to allow it to be used in the signing of the CSRfile.



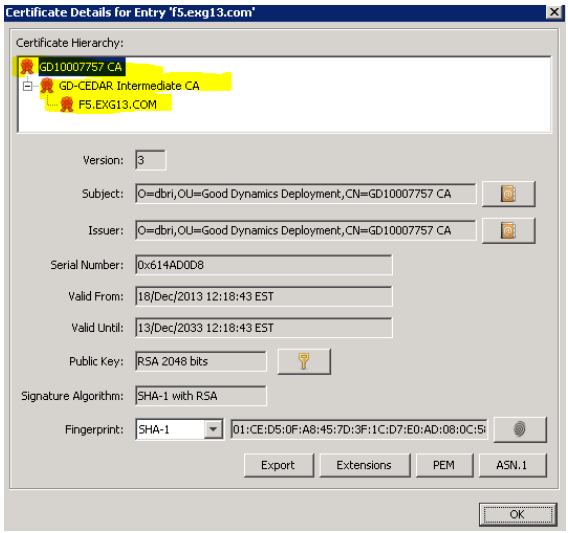
14. Browse to the CSRfile generated, select it, and then fill in the field shown below to have the tool output a CSR reply file. Choose the location and name in the field shown.



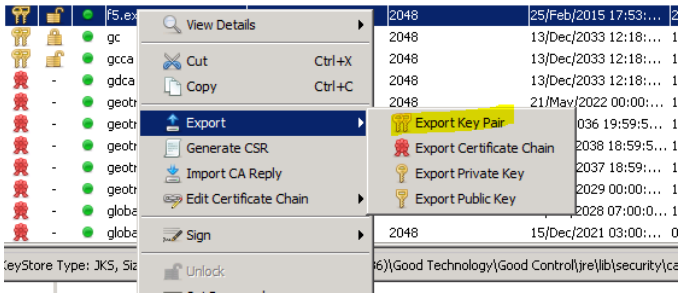
15. Next, right-click on the newly created keypair, and select “Import CA Reply” as shown below. Browse to the saved .p7r file from the previous step and select OK. The certificate is now complete with exportable public and private key.

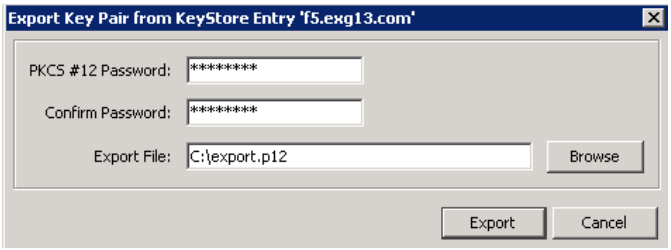


16. By right clicking on the final key-pair you and selecting certificate chain details, you should see details similar to the following screenshot, showing the Root CA, the intermediate CA, and the final certificate you just completed.



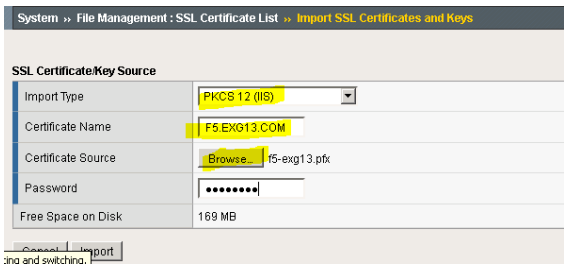
17. Final step is to export the Key Pair and save the .p12 file for import to F5.



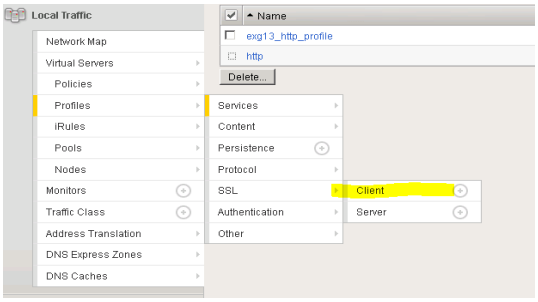


Configuring the F5 client-side SSL profile

1. In the F5 GUI, select System, File Management, SSL Certificate List, Import SSL Certificates and Keys. Browse to the previously export .pfx file, provide a recognizable Certificate Name, the password used to export, and click “import” – this saves the certificate and private key into the F5 repository for use in setting up the server SSL profile.



2. Select Local Traffic, Profiles, SSL, Client – this will allow you to create a client profile for SSL authentication.



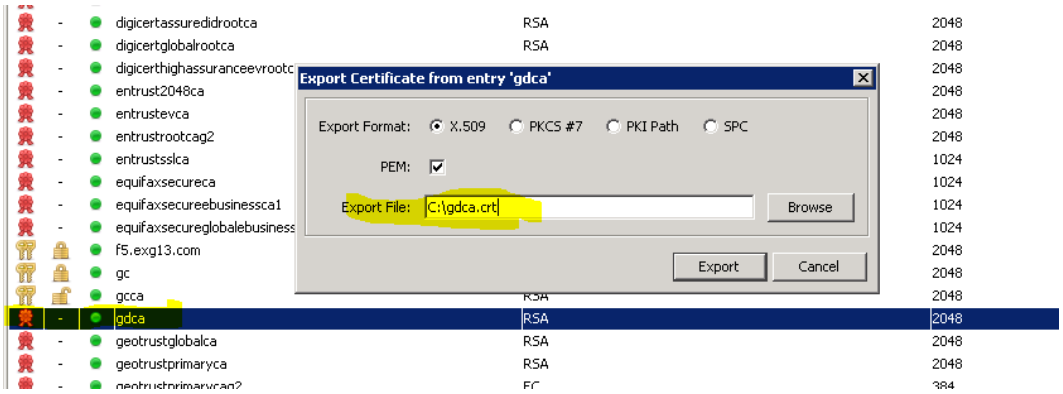
3. Select "Create" in the upper right side of the screen to create a new Client SSL profile.

   Do not change the parent profile clientssl, select the "custom" box on the right, and select the Certificate and Key file that match what was Imported in the previous step.

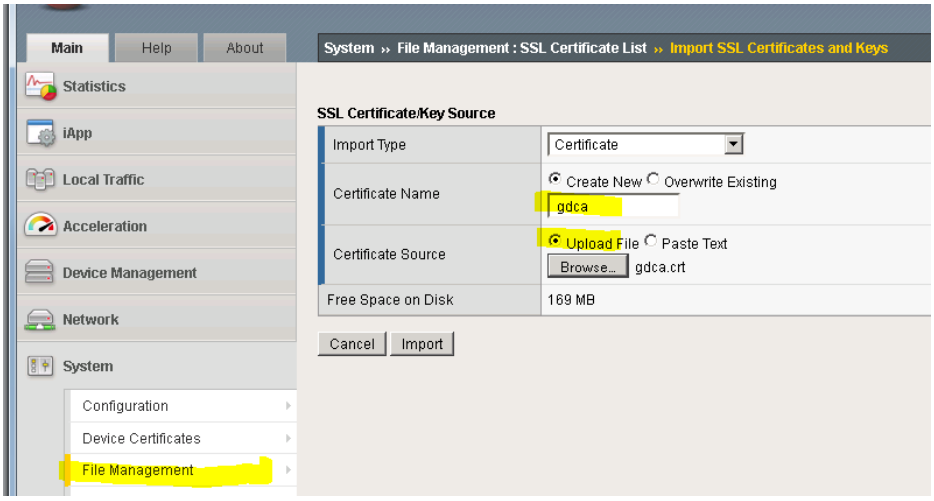
   In this example the name chosen was F5.EXG13-client: All other settings should not be altered on this page.

Configuring the server-side SSL profile

1. In the Keystore Explorer tool, select the gdca public certificate and export to .crt file as shown below.

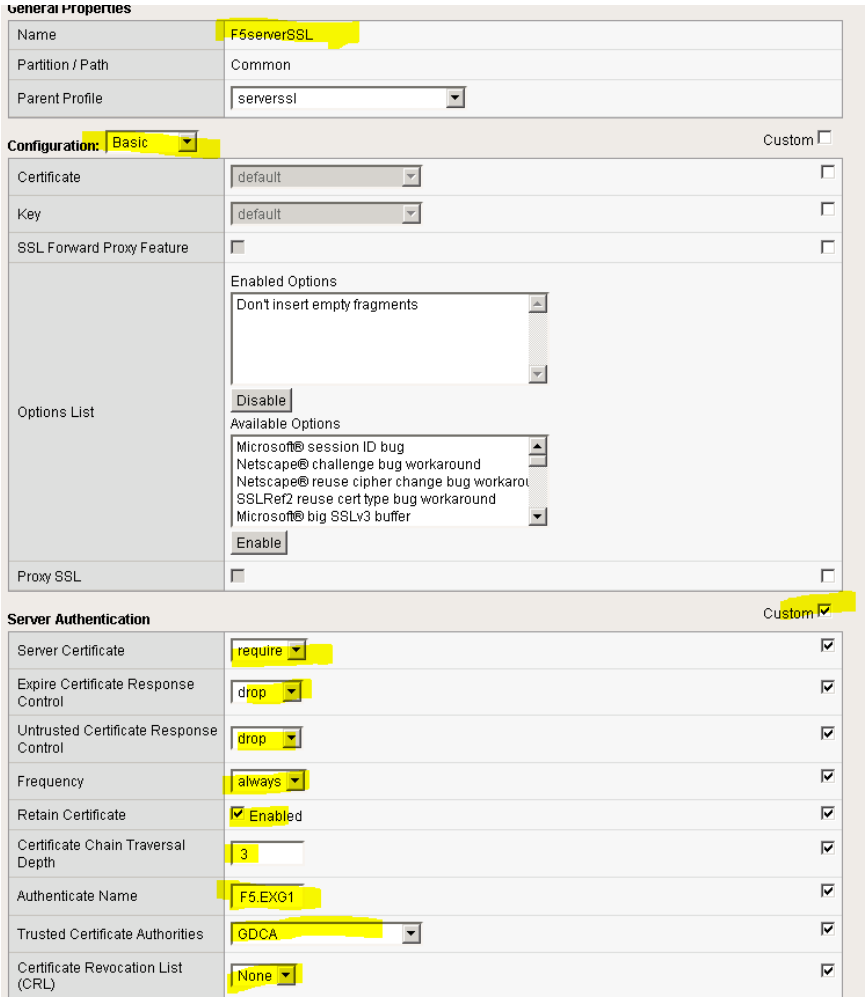


2. Import this file into the F5 SSL certificate store. This certificate is used as the Trusted Certificate Authority instead of the default cacerts bundle included with default F5 profile.



3. Select Local Traffic, Profiles, SSL, Server and create a New Server SSL profile, and name accordingly
4. Modify the server authentication details to include:
   - Server Certificate: Require
   - Expire Certificate Response Control: drop
   - Untrusted Certificate Response control: drop
   - Frequency: once – (can be set to always)
   - Retain Certificate: enabled
   - Certificate Chain traversal depth: 3
   - Authenticate Name – This must be the CN of your created certificate, in this example F5.EXG13.COM
   - Trusted Certificate Authority: GDCA. This must be the certificate you uploaded in a previous step. This allows the F5 to verify the certificate presented by the GP server(s) it establishes connections with to be validated against

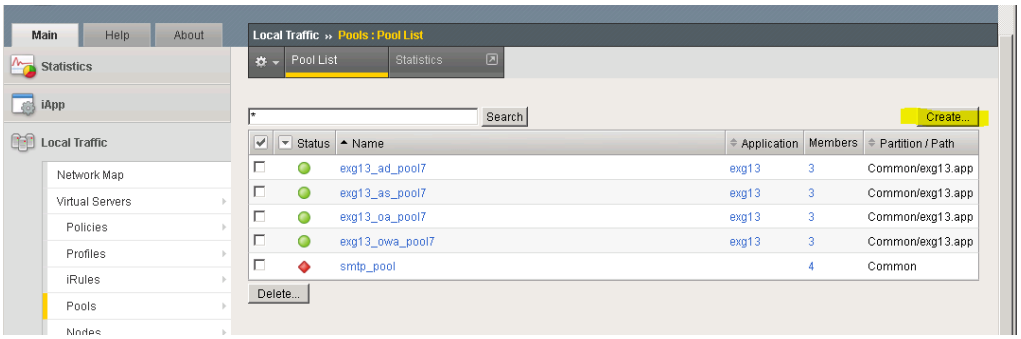
the BlackBerry Dynamics systems proprietary CA.



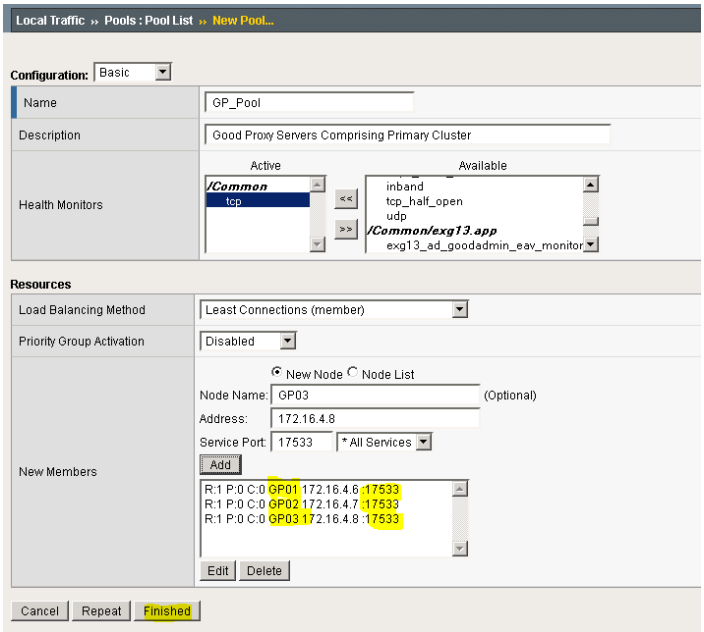
Configuring the F5 server pool

Each member of the GP cluster must also be a member of a pool of servers that F5 will distribute connections to. The method of distribution or load balancing used in this guide is “least-connections” although the actual method the client can choose can vary.

1. From the F5 console, navigate to Local Traffic, Pools, and then select “Create” in the top right.



2. The pool name used in this example is GP_Pool, the health monitor is simple TCP, Load Balancing method “least connections” .
3. Each GP server in the cluster was given identifiable name and associated IP address, along with service port of 17533.
4. Do this for each member of your GP cluster for which the F5 will balance connections.



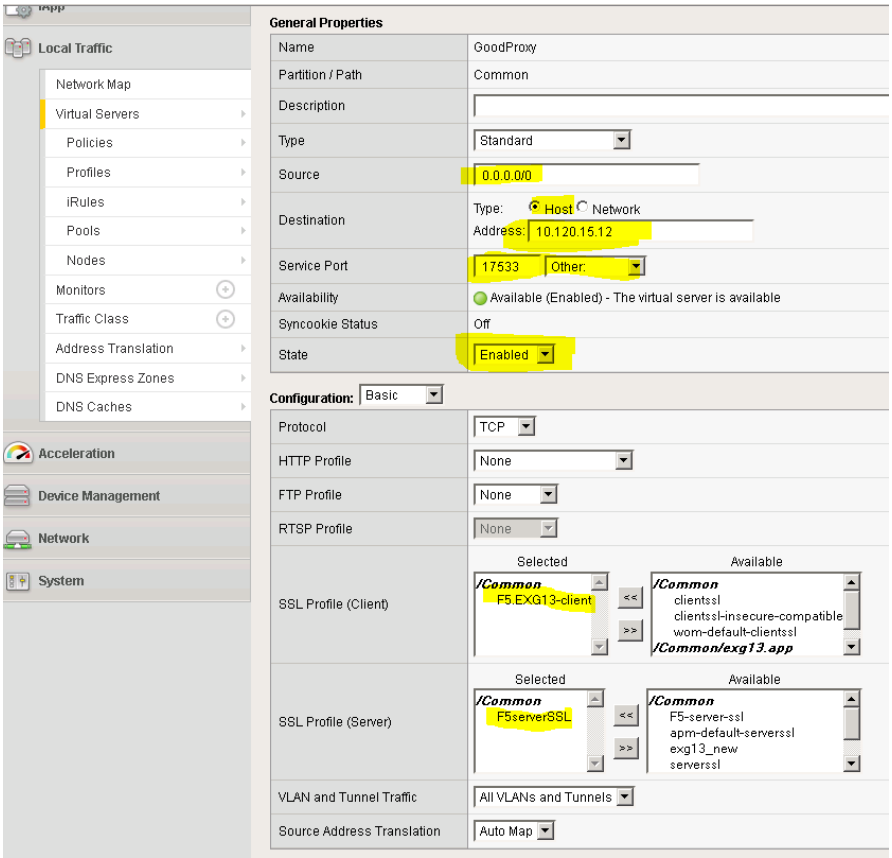
Configuring the F5 virtual server

Note: Except for fields and values specifically called out in these steps, all other values can be left at defaults.

1. From the F5 GUI, go to Local Traffic, Virtual Servers, and select “Create” to create a new Virtual server. This Virtual Server will be the perimeter facing IP address which is NAT’d to from Public IP, or in some cases this could be the actual public IP address which the BlackBerry Dynamics secured applications will make their initial connection to.
2. Source is 0.0.0.0/0 because we will be accepting connections from IP addresses anywhere on the public Internet

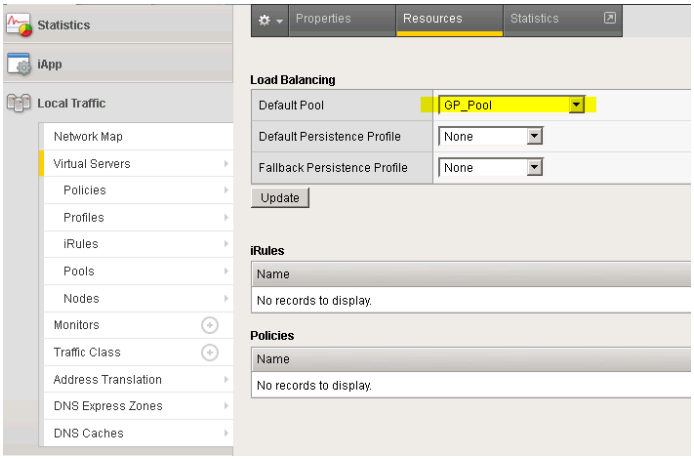
3. Destination will be the perimeter IP address of the F5, in this lab this is the IP address on the internal LAN which is NAT'd to from the Public Interface of Internet Edge Router.
   - Service port must be 17533.
   - Configuration = Basic.
   - Protocol = TCP.
   - SSL Profile (Client) = select the profile created in step 5 above.
   - SSL Profile (Server) = select the custom SSL server profile created in step 6 above.
   - Choose Source Address Translation = Auto-Map (could vary depending on configuration).

> **Note:** HTTP profile should be set to none.



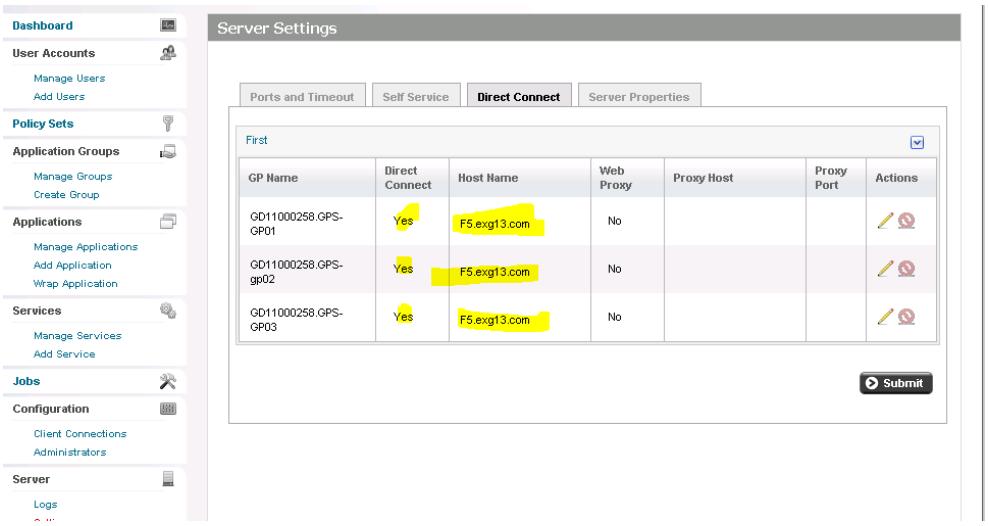
4. Next select Local Traffic, Virtual Servers, GoodProxy (name chosen in previous step), and select "Resources".

5. Select the Default Pool to be associated with this Virtual Server you created previously.



Configuring BlackBerry console settings

1. Any GP server that is a member of a cluster must be configured identically. All members of a cluster must be either Direct Connect enabled or disabled. Broken connections and undesired behavior will result if settings are not uniform.
2. Each member of the GP cluster should be set to Direct Connect = Yes
3. Each member of the GP cluster should have its "Host Name" set to the name identified as the public FQDN of the listener on the F5 reverse proxy – i.e. the Common Name of the Certificate created in the beginning of the configuration.
4. Do not enter anything for the Proxy Host field.



List of supported SSL ciphers between GC and GP servers for Direct Connect

The complete list of supported ciphers is below. These are valid values for the GP server's property file **`c:\good\gps.properties`** and the **`gps.directconnect.supported.ciphers`** key.

SSL_DHE_DSS_EXPORT_WITH_DES40_CBC_SHA

SSL_DHE_DSS_WITH_3DES_EDE_CBC_SHA

SSL_DHE_DSS_WITH_DES_CBC_SHA

SSL_DHE_RSA_EXPORT_WITH_DES40_CBC_SHA

SSL_DHE_RSA_WITH_3DES_EDE_CBC_SHA

SSL_DHE_RSA_WITH_DES_CBC_SHA

SSL_DH_anon_EXPORT_WITH_DES40_CBC_SHA

SSL_DH_anon_EXPORT_WITH_RC4_40_MD5

SSL_DH_anon_WITH_3DES_EDE_CBC_SHA

SSL_DH_anon_WITH_DES_CBC_SHA

SSL_DH_anon_WITH_RC4_128_MD5

SSL_RSA_EXPORT_WITH_DES40_CBC_SHA

SSL_RSA_EXPORT_WITH_RC4_40_MD5

SSL_RSA_WITH_3DES_EDE_CBC_SHA

SSL_RSA_WITH_DES_CBC_SHA

SSL_RSA_WITH_NULL_MD5

SSL_RSA_WITH_NULL_SHA

SSL_RSA_WITH_RC4_128_MD5

SSL_RSA_WITH_RC4_128_SHA

TLS_DHE_DSS_WITH_AES_128_CBC_SHA

TLS_DHE_DSS_WITH_AES_128_CBC_SHA256

TLS_DHE_DSS_WITH_AES_256_CBC_SHA

TLS_DHE_DSS_WITH_AES_256_CBC_SHA256

TLS_DHE_RSA_WITH_AES_128_CBC_SHA

TLS_DHE_RSA_WITH_AES_128_CBC_SHA256

TLS_DHE_RSA_WITH_AES_256_CBC_SHA

TLS_DHE_RSA_WITH_AES_256_CBC_SHA256

TLS_DH_anon_WITH_AES_128_CBC_SHA

TLS_DH_anon_WITH_AES_128_CBC_SHA256

TLS_DH_anon_WITH_AES_256_CBC_SHA

TLS_DH_anon_WITH_AES_256_CBC_SHA256

TLS_ECDHE_ECDSA_WITH_3DES_EDE_CBC_SHA

TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA

TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256

TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA

TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384

TLS_ECDHE_ECDSA_WITH_NULL_SHA

TLS_ECDHE_ECDSA_WITH_RC4_128_SHA

TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA

TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA

TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256

TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA

TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384

TLS_ECDHE_RSA_WITH_NULL_SHA

TLS_ECDHE_RSA_WITH_RC4_128_SHA

TLS_ECDH_ECDSA_WITH_3DES_EDE_CBC_SHA

TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA

TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256

TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA

TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA384

TLS_ECDH_ECDSA_WITH_NULL_SHA

TLS_ECDH_ECDSA_WITH_RC4_128_SHA

TLS_ECDH_RSA_WITH_3DES_EDE_CBC_SHA

TLS_ECDH_RSA_WITH_AES_128_CBC_SHA

TLS_ECDH_RSA_WITH_AES_128_CBC_SHA256

TLS_ECDH_RSA_WITH_AES_256_CBC_SHA

TLS_ECDH_RSA_WITH_AES_256_CBC_SHA384

TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 = **Default**

TLS_ECDH_RSA_WITH_NULL_SHA

TLS_ECDH_RSA_WITH_RC4_128_SHA

TLS_ECDH_anon_WITH_3DES_EDE_CBC_SHA

TLS_ECDH_anon_WITH_AES_128_CBC_SHA

TLS_ECDH_anon_WITH_AES_256_CBC_SHA

TLS_ECDH_anon_WITH_NULL_SHA

TLS_ECDH_anon_WITH_RC4_128_SHA

TLS_EMPTY_RENEGOTIATION_INFO_SCSV

TLS_KRB5_EXPORT_WITH_DES_CBC_40_MD5

TLS_KRB5_EXPORT_WITH_DES_CBC_40_SHA

TLS_KRB5_EXPORT_WITH_RC4_40_MD5

TLS_KRB5_EXPORT_WITH_RC4_40_SHA

TLS_KRB5_WITH_3DES_EDE_CBC_MD5

TLS_KRB5_WITH_3DES_EDE_CBC_SHA

TLS_KRB5_WITH_DES_CBC_MD5

TLS_KRB5_WITH_DES_CBC_SHA

TLS_KRB5_WITH_RC4_128_MD5

TLS_KRB5_WITH_RC4_128_SHA

TLS_RSA_WITH_AES_128_CBC_SHA

TLS_RSA_WITH_AES_128_CBC_SHA256

TLS_RSA_WITH_AES_256_CBC_SHA

TLS_RSA_WITH_AES_256_CBC_SHA256 = **Default**

TLS_RSA_WITH_NULL_SHA256