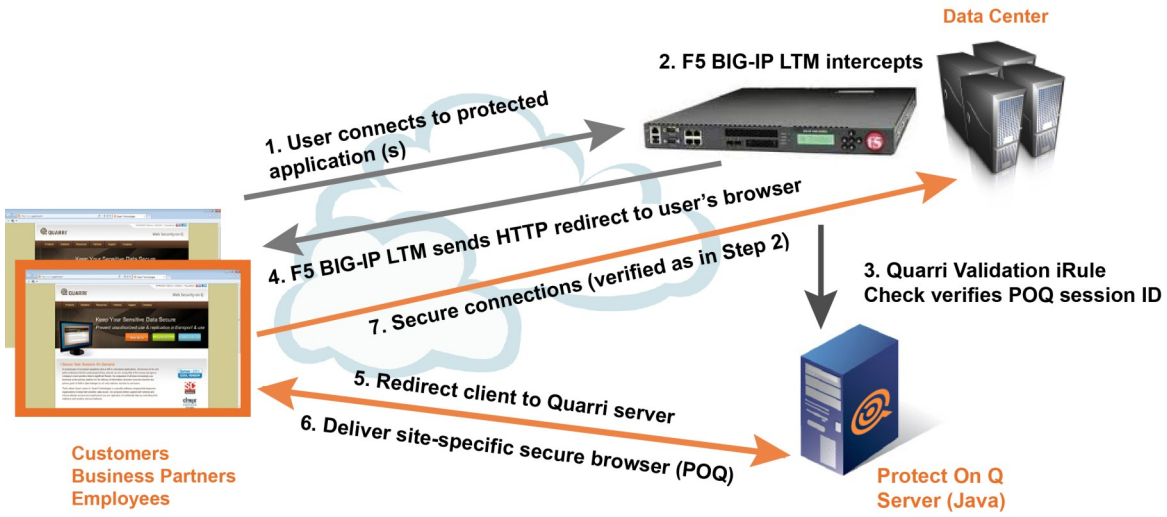
Deploying Quarri Protect On Q with F5 BIG-IP LTM

Welcome to the Quarri Deployment Guide for deploying Quarri™ Protect On Q™ (POQ) with the F5 BIG-IP® Local Traffic Manager™ (LTM). This document is written for organizations that want to integrate POQ with the BIG-IP LTM, allowing the BIG-IP LTM to enforce the use of the protected browser with one or more web applications. This configuration provides unique advantages:

- The BIG-IP LTM is often deployed as a front-end device for multiple independent web applications. This position on the network allows it to require the use of a protected browser with all of these web applications, without modifications to any of the protected web applications.
- The BIG-IP LTM includes features that allow the POQ enforcement iRule to cache POQ session information, minimizing the amount of traffic between the BIG-IP LTM and the POQ Server. This substantially reduces the load on your POQ Server, and minimizes the overhead on end user traffic for POQ enforcement.



Supported Versions

Quarri has two iRule script versions to enforce the use of POQ with BIG-IP LTM:

- One version supports BIG-IP LTM version 10.x
- One version supports BIG-IP LTM versions 11 and later

The two versions of the iRule have different configuration parameters. The parameters that are unique to each version are noted accordingly below.

Contact Quarri to get the right version of the iRule for your BIG-IP LTM.

Preparing the BIG-IP LTM

Before you install and configure the POQ enforcement iRule, you need to do some basic configuration of the LTM.

1. If you haven't already, configure a pool for your protected web application. We'll refer to this pool as the web app pool.
2. Configure a pool for your POQ Server instance(s). We'll refer to this as the POQ pool. If you have more than one POQ Server instance, choose a load balancing scheme that makes sense for your scenario. Sessions do not need to be "sticky" – any HTTP request destined for the POQ Server cluster can be sent to any POQ Server instance.
3. If you haven't already, configure a virtual server for your protected web application. We'll refer to this virtual server as the web app virtual server. This is a public-facing IP address that end users connect to in order to access the web application, and in most cases should be accessed over HTTPS. Set the Default Pool to the web app pool you created above, and set SNAT Pool to "Auto Map".
4. Configure a virtual server for the Protect On Q Server. We'll refer to this virtual server as the POQ virtual server. Set the Default Pool to the POQ pool you created above, and set SNAT Pool to "Auto Map". Normally, users should not be able to access this virtual server directly, so use HTTP (not HTTPS), and limit connectivity to it (e.g., by setting VLAN and Tunnel Traffic to "Enabled on..." and leaving the Selected list empty).

Note:

The POQ Server uses an HTTP redirect during the POQ protected browser launch sequence. Since the traffic between the LTM and the POQ Server is HTTP (not HTTPS), this redirect can be misinterpreted by the LTM, and the user can end up being redirected to a non-HTTPS URL. To avoid this issue, the virtual server handling traffic between end users and the POQ Server should be configured with an ***HTTP Profile*** *that has* ***Redirect Rewrite*** *set to "All". If this*

*setting is inconsistent with how other web resources on this virtual server need to be configured, you may need to configure a separate virtual server to handle traffic between end users and the POQ Server, on a different IP address and/or port. If you do this, you'll need to update the **static::launch_uri** parameter in the iRule to match the POQ Server's virtual server address (from the users' perspective).*

Installing the iRule

Now you're ready to import and configure the iRule script. There are several ways to add and edit iRules, including the BIG-IP LTM web-based administrative console, and the F5 iRule Editor, a standalone program available for download from F5. If you don't have a copy of the POQ enforcement iRule, please contact support@quarri. com.

1. Create a new iRule. Give it a name like "quarri_require_poq", and copy/paste the iRule contents.
2. Configure the iRule's static parameters. These are found at the top of the iRule script, in the "when RULE_INIT" section.
    3. **static::debug** – Whether to write debug log messages to /var/log/ltm. Normally should be disabled (0).
    4. **static::policy_name (LTM 10 only)** – The name of the policy users should be using. This is used in validating users' session keys.
    5. **static::session_timeout_minutes (LTM 10 only)** – The session timeout defined in your POQ policy. This value must match the value from the policy, or users sessions will be expired too early or too late.
    6. **static::server_shared_secret (LTM 10 only)** – The shared secret from the POQ Server's configuration file. For more details on how to find/set this, see the POQ Administrator's Guide.
    7. **static::policy_shared_secret (LTM 10 only)** – The shared secret from the policy. You can obtain this value by opening the policy definition in the POQ Manager, and looking in the Advanced tab.
    8. **static::caching (LTM 11+ only)** – Whether caching is enabled. For performance reasons, caching should normally be enabled (1).
    9. **static::dyn_timeout (LTM 11+ only)** – Whether to use the Expires header value from the POQ Server response to determine session key cache timeout. Normally should be enabled (1).
    10. **static::cache_timeout (LTM 11+ only)** – Static timeout for cached session keys; only used if dyn_timeout is disabled.

11. **static::launch_uri** – The launch URI for the POQ policy you're using, from an end user's perspective. This needs to be consistent with the virtual server address/port that will be handling traffic between end users and the POQ Server.
12. **static::poq_virtual_server** – The name of the virtual server representing the POQ Server (the POQ virtual server you configured above), which is called by the iRule to validate session keys for end users' HTTP requests.
13. **static::pass_poqserver_requests** – If the POQ Server URLs (like /poqserver/*) are hosted on the same virtual server as protected web sites, turn this on to allow requests to the POQ Server to pass without a POQ session key (so users can launch the protected browser).
14. **static::pass_poqmanager_requests** – If the POQ Manager URLs (like /poqmanager/*) are hosted on the same virtual server as protected web sites, turn this on to allow requests to the POQ Manager to pass without a POQ session key. Normally it is a BAD IDEA to expose the POQ Manager on internet-facing network interfaces, for security reasons.
15. **static::poqserver_pool** – If you're using pass_poqserver_requests, define what pool such requests will be sent to here. In that case, this should be set to the name of the POQ pool you created above.
16. **static::poqmanager_pool** – If you're using pass_poqmanager_requests, define what pool such requests will be sent to here.
17. **static::error_html_header** – HTML template header for error messages.
18. **static::error_html_footer** – HTML template footer for error messages.

19. Save the new iRule.
20. Attach the iRule to the webapp virtual server from above. You can do this on the Resources tab of the virtual server definition.
21. Try to access an application on the webapp virtual server from an unprotected browser, and you should be redirected to the POQ launch page. When the protected browser opens, the page will be the URL you were originally trying to visit.

Quarri Technologies, Inc.
7500 Rialto Blvd.
Building 2, Suite 210
Austin, TX 78735

1.866.248.3990 US
+1.512.590.7731
+1.512.777.5005 Fax

info@quarri.com
www.quarri.com