Running RMsis on Reverse Proxy with JIRA Data Center

Overview

RMsis must be configured to run behind a reverse proxy server like Nginx or Apache HTTP Server. Configuring reverse proxy server allows for running RMsis on non-standard HTTP/HTTPS port (such as 3060) and users will be able to access RMsis over standard HTTP/HTTPS as their traffic will be routed through the proxy.

This page describes how to integrate Apache HTTP Server with RMsis, utilizing mod_proxy so that Apache operates as a reverse-proxy over HTTPS (You can use similar/ appropriate configuration if you are using Nginx).

Target Audience :

The targeted audience for this document are users who want to

- access RMsis over HTTPS but do not want to install/ configure security certificate.
- run RMsis 2.x DC with JIRA DC using a reverse proxy server/load balancer.

Prerequisites :

The following are the prerequisites for running RMsis 2.x with JIRA DC using a reverse proxy server :

- A reverse proxy server (like Apache HTTP Server or Nginx) should be installed and running.
- JIRA DC installation should be running on Reverse Proxy configuration using Apache HTTP Server. Jira Data Center Load Balancer examples for integrating Apache HTTP Server reverse proxy with JIRA DC is available here: https://confluence.atlassian.com/enterprise/jira-data-center-load-balancer-examples-781200827.html

A typical Reverse Proxy/Load balancer Configuration for RMsis 2.x DC with JIRA DC

If the above-mentioned prerequisites are fulfilled, a typical reverse Proxy Configuration with JIRA and RMsis running on HTTP will look like :

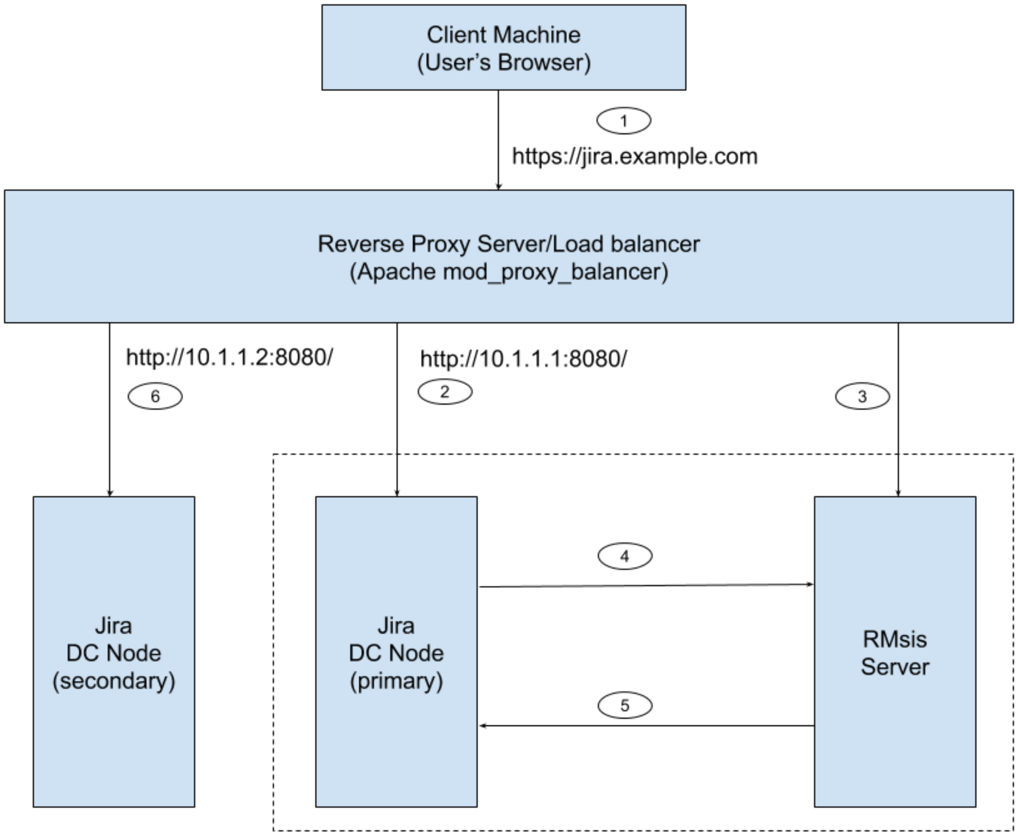


Figure 1 : A typical Reverse Proxy Configuration

In the above image, the reverse proxy server is configured for the URL https://jira.example.com , the JIRA DC nodes are configured for URL 10.1.1.1 and 10.1.1.2

Link #1 : Client machine accessing reverse proxy server - https://jira.example.com

Link #2 : Reverse Proxy Server accessing JIRA DC Node(primary) on which RMsis is being deployed: http://10.1.1.1:8080/

Link #6 : Reverse Proxy Server accessing remaining node of JIRA DC(secondary) : http://10.1.1.2:8080/

Now, We need to configure links #3, and #5 (Link #4 is automatically created by RMsis).

Configure Reverse Proxy Server (for RMsis)

Apache uses Modules which can be enabled or disabled and are essentially plugins that change the functionality of the server .mod_proxy is an example of these modules.

To configure `mod_proxy` for use with your Application, you need to use the `ProxyPass` and `ProxyPassReverse` directives in the Apache configuration file as follows:

```
ProxyPass /rm http://jira.example:3060/rm
ProxyPassReverse /rm http://jira.example:3060/rm
```

The above directives tell Apache HTTP Server to forward web requests of the form `https://jira.example.com/rm` to the Tomcat connector running on port `3060`(in this case RMsis running under the context path /rm) on the same machine.

Detailed information for mod_proxy is available here: https://httpd.apache.org/docs/2.4/mod/mod_proxy.html

If you are using some other reverse proxy server (like Nginx), you can use similar configuration/ modules.

Sample Reverse Proxy Configuration for Apache :

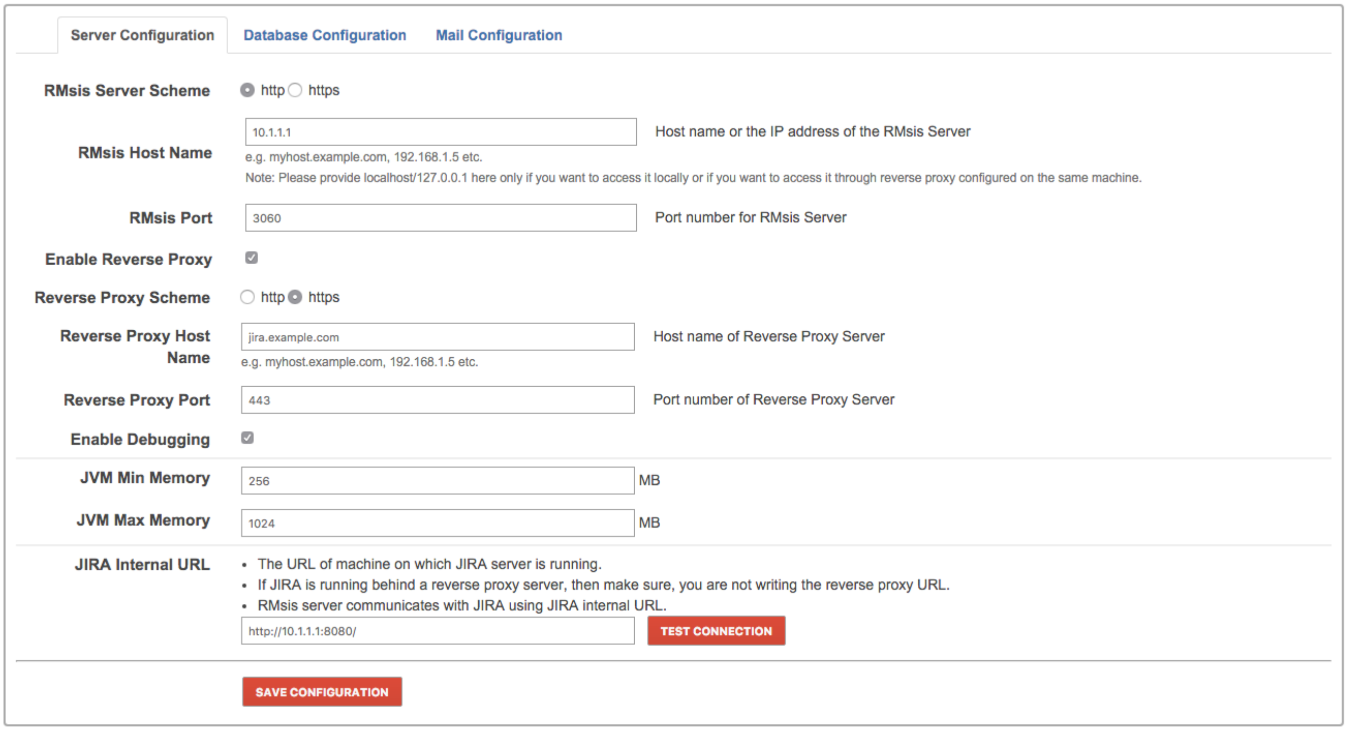
If JIRA DC cluster is running under the root context "/",then the sample reverse proxy configuration for both JIRA DC and RMsis 2.x DC servers would be:

```
NameVirtualHost *:443

<VirtualHost *:443>
    ServerName jira.example.com

    #The above link refers to Link #1 in Figure 1

ProxyRequests Off
    ProxyPreserveHost On


    SSLEngine On
    SSLCertificateFile /home/user/ssl/server.crt
    SSLCertificateKeyFile /home/user/ssl/server.key
    SSLCertificateChainFile /home/user/ssl/cabundle.crt


        #for JIRA DC Cluster Setup

        <Proxy balancer://jiracluster>
                # JIRA DC node 1
                BalancerMember http://10.1.1.1:8080 route=primary
                # The above link refers to Link #2 in Figure 1

                # JIRA DC node 2
                BalancerMember http://10.1.1.2:8080 route=secondary
                #The above link refers to Link #6 in Figure 1

                # Security "we aren't blocking anyone but this the place to make those changes
                Order Deny,Allow
                Deny from none
                Allow from all

                # Load Balancer Settings
                # We are not really balancing anything in this setup, but need to configure this
                ProxySet lbmethod=byrequests
                ProxySet stickysession=JSESSIONID
        </Proxy>


    #for RMsis 2.x DC Server
            ProxyPass /rm http://10.1.1.1:3060/rm
            ProxyPassReverse /rm http://10.1.1.1:3060/rm

            #The above link refers to Link #3 in Figure 1


         #for JIRA Data Center
        # Don't reverse-proxy requests to the management UI
        ProxyPass /balancer-manager !

        # Reverse proxy all other requests to the JIRA cluster
        ProxyPass / balancer://jiracluster/

        # Here's how to enable the load balancer's management UI if desired
        <Location /balancer-manager>
                SetHandler balancer-manager

                # You SHOULD CHANGE THIS to only allow trusted ips to use the manager
                Order deny,allow
                Allow from all
        </Location>

</VirtualHost>
```

Note : The reverse proxy directives for RMsis should be written **before the reverse proxy directives for JIRA DC** in the Apache configuration file if JIRA is running under the root context / .

Configure RMsis Server :

The server configuration in RMsis will be :



RMsis Server Scheme: ● http ○ https

RMsis Host Name: 10.1.1.1 — Host name or the IP address of the RMsis Server
e.g. myhost.example.com, 192.168.1.5 etc.
Note: Please provide localhost/127.0.0.1 here only if you want to access it locally or if you want to access it through reverse proxy configured on the same machine.

RMsis Port: 3060 — Port number for RMsis Server

Enable Reverse Proxy: ☑

Reverse Proxy Scheme: ○ http ● https

Reverse Proxy Host Name: jira.example.com — Host name of Reverse Proxy Server
e.g. myhost.example.com, 192.168.1.5 etc.

Reverse Proxy Port: 443 — Port number of Reverse Proxy Server

Enable Debugging: ☑

JVM Min Memory: 256 MB

JVM Max Memory: 1024 MB

JIRA Internal URL:

- The URL of machine on which JIRA server is running.
- If JIRA is running behind a reverse proxy server, then make sure, you are not writing the reverse proxy URL.
- RMsis server communicates with JIRA using JIRA internal URL.

http://10.1.1.1:8080/

- **RMsis Server Scheme: http** (scheme on which RMsis server will be running)
- **RMsis Host Name: 10.1.1.1** (Host name or the IP address of the RMsis Server)
- **RMsis Port: 3060** (Port number for RMsis Server)
  - The above three will create Link #3 in Figure 1
- **Enable Reverse Proxy: Enabled**
- **Reverse Proxy Scheme: https** (scheme on which Reverse Proxy Server will be running)
- **Reverse Proxy Host Name: jira.example.com** (Host name of Reverse Proxy Server)
- **Reverse Proxy Port: 443** (Port number of Reverse Proxy Server)
- **Enable Debugging: Enabled**
- **JVM Min Memory: 256 MB**
- **JVM Max Memory: 1024 MB**
- **JIRA Internal URL: http://10.1.1.1:8080/**
  - This creates Link #5 in Figure 1.
  - This is the URL of the machine on which JIRA+RMsis server is running.
  - Make sure that you are not writing the reverse proxy URL here.
  - RMsis server communicates with JIRA using JIRA internal URL.

The complete Reverse proxy configuration will look like :

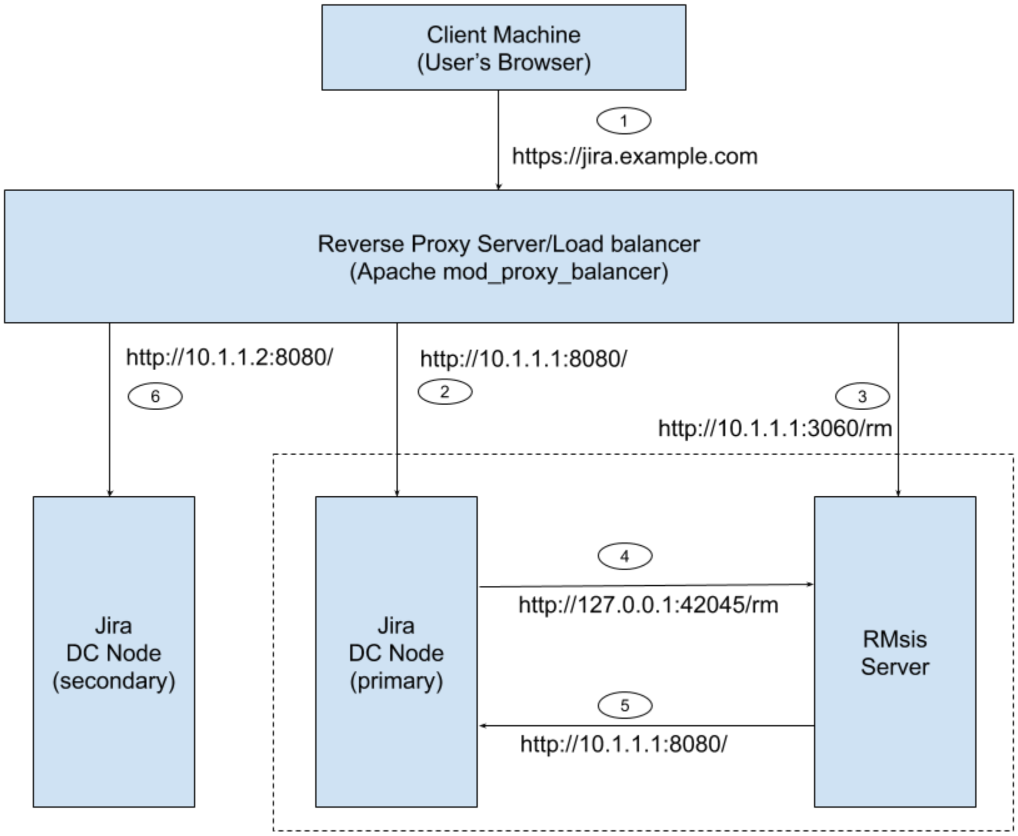


Figure 2 : Reverse Proxy Configuration

Link #1 : Client machine accessing reverse proxy server - https://jira.example.com

Link #2 & #6 : Reverse Proxy Server accessing JIRA DC Nodes : http://10.1.1.1:8080/ & http://10.1.1.2:8080/

Link #3 : Reverse Proxy Server accessing RMsis 2.x DC Server : http://10.1.1.1:3060/rm

Link #4 : JIRA Server accessing RMsis Server : http://127.0.0.1:42045/rm (here 42045 is an internal port selected automatically by RMsis. If this port is not available, RMsis will select a different port)

Link #5 : RMsis Server accessing JIRA Server : http://10.1.1.1:8080/

Note :

For this configuration to work correctly, reverse proxy should not be blocking access from

- JIRA to RMsis
- RMsis to JIRA

Further references:

A guide to reverse proxy is available here: https://httpd.apache.org/docs/2.4/howto/reverse_proxy.html