| **File name** | YTM Enterprise Deployment Appliance |
| --- | --- |
| **Author** |  |
| **Confidentiality** | Internal |
| **Last save date** | Tuesday, December-01-2020 at 3:23:00 PM |

Table of Contents

[1 Introduction 2](#_Toc57735876)

[2 Enterprise Configurations 2](#_Toc57735877)

[2.1 High Performance Solution 2](#_Toc57735878)

[2.2 High Performance and High Availability Solution 3](#_Toc57735879)

[3 Apache HTTP Server 4](#_Toc57735880)

[3.1 Step by Step Configuration Guide 4](#_Toc57735881)

[3.2 Apache HTTPD and MOD\_JK Installation 4](#_Toc57735882)

[3.3 MOD\_JK Configuration 5](#_Toc57735883)

[3.4 Tomcat Setup 6](#_Toc57735884)

[3.5 Open Issues 6](#_Toc57735885)

[4 Useful links 6](#_Toc57735886)

[5 PostgreSQL Hot Standby Configuration 6](#_Toc57735887)

[6 Limitations and Concerns 8](#_Toc57735888)

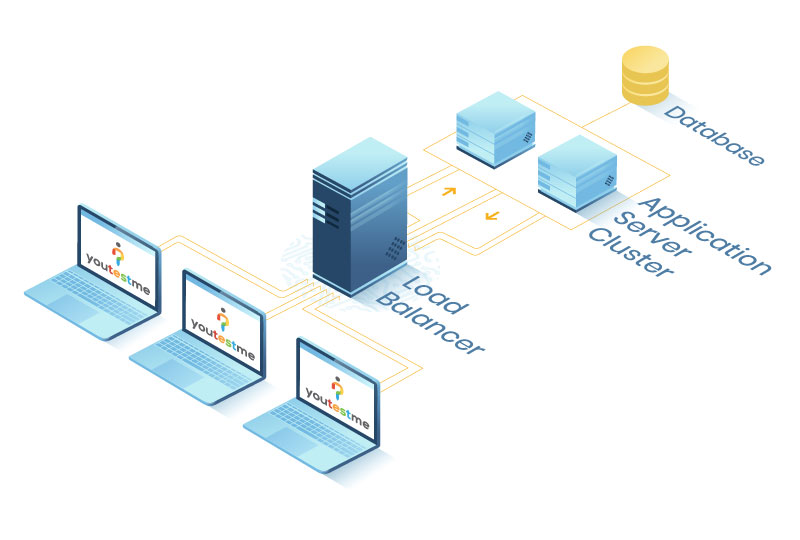
# Introduction

This document describes the configuration of the YTM Enterprise Deployment Appliance.

# Enterprise Configurations

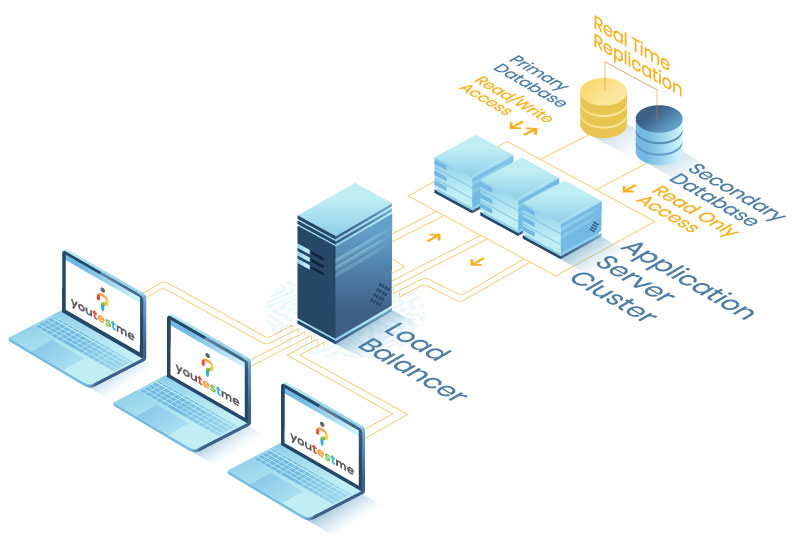
## High Performance Solution

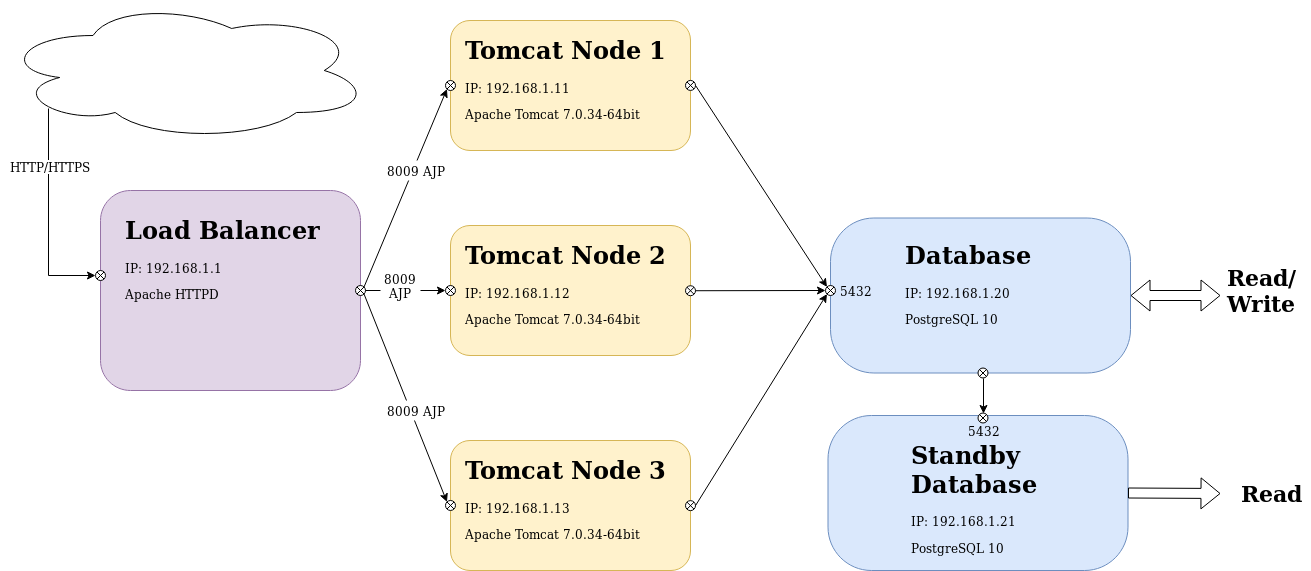
| **#** | **Layer** | **Min Number of VMs**  **Required** | **Description** |
| --- | --- | --- | --- |
|  | Apache HTTP Server  https://httpd.apache.org/docs/2.4/ | 1 | v2.4 |
|  | A number of Tomcat Servers (default 2) | 2+ | Apache tomcat 7.0.34-64bit |
|  | PostgreSQL database server | 1 | v10 |



## High Performance and High Availability Solution

| **#** | **Layer** | **Min Number of VMs**  **Required** | **Description** |
| --- | --- | --- | --- |
|  | Apache HTTP Server | 1 | v2.4 |
|  | A number of Tomcat Servers (default 2) | 2+ | Apache tomcat 7.0.34-64bit |
|  | A number of PostgreSQL servers (default 2) | 2 | Primary and Hot Standby database - v10 |





# Apache HTTP Server

## Step by Step Configuration Guide

<https://www.mulesoft.com/tcat/tomcat-clustering>

## Apache HTTPD and MOD\_JK Installation

Install HTTPD package with other useful packages by running the following command:

# yum install httpd-devel apr apr-devel apr-util apr-util-devel gcc gcc-c++ make autoconf libtool

Download and install Tomcat connector “mod\_jk” from the official site:

<http://tomcat.apache.org/download-connectors.cgi>.

# mkdir -p /opt/mod\_jk/

# cd /opt/mod\_jk

# wget http://www.eu.apache.org/dist/tomcat/tomcat-connectors/jk/tomcat-connectors-1.2.41-src.tar.gz

# tar -xvzf tomcat-connectors-1.2.41-src.tar.gz

# cd tomcat-connectors-1.2.41-src/native

# ./configure --with-apxs=/usr/sbin/apxs --enable-api-compatibility

# make

# libtool --finish /usr/lib64/httpd/modules

# make install

## MOD\_JK Configuration

Open the file “/etc/httpd/conf/httpd.conf” and add the following:

LoadModule jk\_module "/etc/httpd/modules/mod\_jk.so"

JkWorkersFile /etc/httpd/conf/workers.properties

# Where to put jk shared memory

JkShmFile /var/run/httpd/mod\_jk.shm

# Where to put jk logs

JkLogFile /var/log/httpd/mod\_jk.log

# Set the jk log level [debug/error/info]

JkLogLevel info

# Select the timestamp log format

JkLogStampFormat "[%a %b %d %H:%M:%S %Y] "

<Location /status>

JkMount jkstatus

Require ip [Allowed\_IP]

</Location>

<IfModule mod\_jk.c>

JkMount /\* LoadBalancer

</IfModule>

Create new directory to store shared memory of the module:

**# mkdir -p /var/run/httpd/mod\_jk**

**# chown apache:apache /var/run/httpd/mod\_jk**

Next step will be to create “worker.properties” file with the following content:

**# vim /etc/httpd/conf/workers.properties**

worker.list=jkstatus, LoadBalancer

worker.jkstatus.type=status

worker.LoadBalancer.type=lb

worker.jvm1.type=ajp13

worker.jvm1.host=[TOMCAT1\_IP]

worker.jvm1.port=[TOMCAT1\_AJP\_PORT]

worker.jvm1.lbfactor=1

worker.jvm2.type=ajp13

worker.jvm2.host=[TOMCAT2\_IP]

worker.jvm2.port=[TOMCAT2\_AJP\_PORT]

worker.jvm2.lbfactor=1

worker.LoadBalancer.balance\_workers=jvm1,jvm2

worker.LoadBalancer.sticky\_session=true

## Tomcat Setup

The “jvmRoute” attribute of the Engine element allows the load balancer to match requests to the JVM currently responsible for updating the relevant session. It does this by appending the name of the JVM to the JSESSIONID of the request, and matching this against the worker name provided in “workers.properties”.

In order to configure *jvmRoute*, make sure that the value of “jvmRoute” for all your Engines is paired with an identically named Worker name entry in mod\_jk’s worker.properties configuration file.

We should set “jvmRoute” to support load-balancing via AJP by modifying Engine element:

<Engine name=”Catalina” defaultHost=”localhost” jvmRoute=”jvm1”>

At the end of the configuration process it is necessary to restart Tomcat server and Apache HTTPD.

## Open Issues

1. **LB keep sending connections to worker marked as disabled**

The issue occurs on testing in the company. After testing in LoadBalancing mode (both applications active), status for one worker is configured as disabled, and apache HTTPD is restarted using command *graceful.* After that, all new connections should be sent to one worker, but they are still sent to the disabled one.

# Useful links

1. Configuring tomcat cluster with load balancer - <https://www.mulesoft.com/tcat/tomcat-clustering>
2. Official tomcat workers document - <http://tomcat.apache.org/connectors-doc/common_howto/loadbalancers.html>
3. Official tomcat cluster document - <https://tomcat.apache.org/tomcat-8.5-doc/cluster-howto.html>
4. Example with session replication in Java - <https://examples.javacodegeeks.com/enterprise-java/tomcat/tomcat-clustering-session-replication-tutorial/>
5. Tomcat Clustering configuration tutorial:

<http://www.datadisk.co.uk/html_docs/java_app/tomcat6/tomcat6_clustering.htm>

1. <https://geekflare.com/tomcat-load-balancer-using-mod-proxy-and-session-sticky/>
2. [https://access.redhat.com/documentation/mod\_jk](https://access.redhat.com/documentation/en-us/red_hat_jboss_web_server/3/html/http_connectors_and_load_balancing_guide/sect-configure_load_balancing_using_apache_http_server_and_mod_jk)

# PostgreSQL Hot Standby Configuration

All relevant information can be found in the following document:

<https://svn.youtestme.com/admin/trunk/System> Administration/PostgreSQL/YTM PostgreSQL Replication and Hot Standby.docx

# Limitations and Concerns

| **Report Location** | **Summary** |
| --- | --- |
| [https://svn.youtestme.com/scm/svn/qa/trunk/YTM Group Tests/Group Tests Reports/YTM Limitations and Concerns of EE Configuration 01.12.2020..docx](https://svn.youtestme.com/scm/svn/qa/trunk/YTM%20Group%20Tests/Group%20Tests%20Reports/YTM%20Limitations%20and%20Concerns%20of%20EE%20Configuration%2001.12.2020..docx) | **System configuring does not apply to all applications on EE.**  **1. Proctoring service is set up on the application with VM 192.168.1.114**  **the user who is not logged in to the same application will not be able to start the proctored test.**  **2. Pause the session only works for users at the same VM IP address as the admin**. |