

**On-Premise Deployment**

September 28, 2016

**Document Control Information**

**Document Edit History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | Date | Description | Author |
| 1.0 | 08/31/2016 | Initial Version | Ankit Prakash |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Support / SME Resources**

|  |  |  |  |
| --- | --- | --- | --- |
| **Team** | Name | Role | Contact (Email / Phone) |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

[1. Overview 4](#_Toc460424522)

[2. Purpose 4](#_Toc460424523)

[3. List of Servers 4](#_Toc460424524)

[4. Deployment Process 4](#_Toc460424525)

[5. Common Reasons for Deployment Failure 5](#_Toc460424526)

1. Overview

This document provides the On Premise deployment process which are required to understand

for doing the deployments on the TEST and PROD servers.

1. Purpose

This document talks about the complete procedure and processes of the code deployment on the

on Premise server Atlanta and Tulsa. This document discusses about name of the server, complete

deployment of the zip which should be taken from svn, connecting to the remote server, mule

server stop starts commands which are required from deploying the zip code and facing issues

while deploying the zip code on the on premise server.

1. List of Servers

**TEST**

USATLT150.kosaz.com (ATLANTA)

USTULT150.kosaz.com (TULSA)

**PROD**

USATLB150.kosaz.com (ATLANTA)

USTULB150.kosaz.com (TULSA)

1. Deployment Process

Steps need to be follow while doing the deployment on OnPremise Server.

**Step 1:-** Login to VPN and Check-out the source code from SVN which is required to deploy on

server.

**Step 2:-** To deploy the application from source code, open the project in Anypoint Studio

IDE and import as Maven Project using POM.xml.

**Step 3:-** Add the missing or dependency jars if required and run it locally.

**Step 4:-** If the local build is successful then export the project as .zip or take the .zip file

from target folder and check-in, the build file with new version into the SVN.

**Step 5:-** Open the browser and keep the track of the build file location in SVN.

**Step 6:-** Connect to Remote Server using Remote Desktop Connection. Give the server name,

username and password.

**e.g.** servername = USATLT150.KOSAZ.COM

username = kosaz\koID

password = \*\*\*\*\*\*\*

**Step 7:-** Paste the tracked url location of build file on the Mozilla browser. Download the

build file and save the file to downloads.

**Step 8:-** Copy the file to ‘C’ folder/drive.

**Step 9:-** If the deployment already exists and new file came for deployment of the same,

we need to make sure build file names are same. For this rename the build file

name as previous build file name.

**Step 10:-** Then copy the file to apps folder then check the deployment happens or not. If

it happened then ok otherwise we need to restart the server.

**Step 11:-** Now Stop the mule server using ‘mule stop’ command from windows command

line console.

**Step 12:-** Now start the server using the ‘mule start <parameters>’ command from

windows command line console.

Parameters right now are

[-M-Dsap.inbound.mule.env=test [-M-Dsap.inbound.key=cokewdsit@16coke](mailto:-M-Dsap.inbound.key=cokewdsit@16coke)]

**Step 13:-** If it deploys successfully it will create the anchor file, if it failed to create the

anchor file, please send the corresponding log file from logs folder to the

developer. So they can fix the issue.

**Step 14:-** If the deployment is stuck and started server is not deploying any other app

we need to do the following

**#** Go to google and search for killing windows service.

**#** Use that details to kill windows service mule enterprise edition 3.7.1

**#** Open task manager and kill the java process that is got 0% CPU utilization.

1. Common Reasons for Deployment Failure

There are some of the reasons, why successfully build code (in local system) fails on premise

deployment.

**1).** Forget to delete the http connecter, which is meant for development testing.

**2).** Java version conflict. Suppose the code is build pn JDK 1.8, but while deploying on

server JDK 1.7 or lower version is there, then the build deployment will fail. We need

to make sure that if OnPremise server have JDK 1.7.0\_79 then the code should be

compiled with the same version(not even JDK 1.7.0\_80 compiled code will work it

will give the major-minor version error on server).

**3).** Sometimes in application we keep ${mule.env} in configuration file and we declared this

property in mule.app.properties, in this case also the deployment will fail. So we

shouldn’t read anything from mule.app.properties in application code. This happens

also for Cloud deployments.