



SAS Life Science Analytics Framework Java API 2.2

Installation Qualification / Operational Qualification Checklist

Install LSAF Java API 2.2

Version 1.00
Date October 2019

Checklist A: "Install LSAF Java API"

Note: This checklist assumes you have completed an installation of LSAF 5.1.x and are installing the LSAF Java API.

Checklist A: "Install LSAF Java API"		Corresponding Document: <i>None</i>	
#	Activity	Expected Results	Actual Results
A1.	Login to an admin PC.	Windows display appears.	
A2.	Using the putty tool, create an ssh connection from the admin PC to the web server and login as webtrust .	You're prompted for the password.	
A3.	Enter the <i>password</i> for webtrust.	You're logged into web server.	
A4.	Use the script command to capture commands that are typed in the Unix window. The syntax is script /var/staging_logs/<servername>_lsafapi2.2_<date>.txt	System should indicate script has started.	
A5.	From the admin PC, download the LSAF Java API 2.2 distribution from the SAS Technical Support site. After the distribution has been downloaded, use the WinSCP2 tool to copy the file to /home/webtrust/Installs on the web server. Transfer it as a binary file using the webtrust userid. lsaf-java-api-server-2.2.zip <i>Note:</i> In a clustered web server environment, this file needs to be copied to each web server in the cluster.	The file is transferred to the web server.	
A6.	Type cd /sso/sfw/tcServer	Navigate to that directory.	

Checklist A: "Install LSAF Java API"		Corresponding Document: <i>None</i>	
#	Activity	Expected Results	Actual Results
A7.	Type <code>./tcruntime-ctl.sh lsafserver1 stop</code>	System displays: Instance is running as PID=29796, shutting down... Instance is running PID=29796, sleeping for up to 60 seconds waiting for shutdown Instance shut down gracefully	
A8.	Type <code>ps -ef grep webtrust</code>	You should not see the tcserver process running.	
A9.	Type <code>cd lsafserver1/logs</code>	Navigate to the <code>/sso/sfw/tcServer/lsafserver1/logs</code> directory.	
A10.	Delete or rename the <code>lsaf_info.log</code> and <code>catalina.out</code> files.	Log files are deleted or renamed.	
A11.	Type <code>ls -al</code>	Verify that log files have been renamed or deleted.	
A12.	Repeat steps A6 – A11 for the remaining tcServers. <i>Note:</i> You do not have to stop the <code>lsafserveradmin</code> server. <i>Note:</i> In a clustered web server environment, steps A6 – A12 must be performed on each web server in the cluster.	The tcServers are stopped except for the <code>lsafserveradmin</code> server.	
A13.	Type <code>cd /home/webtrust/Installs</code>	Navigate to that directory.	
A14.	Type <code>unzip lsaf-java-api-server-2.2.zip</code>	The contents of the zip file are extracted.	
A15.	Type <code>ls -al</code>	You will see a <code>lsaf-java-api-server-2.2</code> directory that was created in the previous step.	
A16.	Type <code>cd lsaf-java-api-server-2.2</code>	Navigate to that directory.	

Checklist A: "Install LSAF Java API"		Corresponding Document: <i>None</i>	
#	Activity	Expected Results	Actual Results
A17.	Type ant clean	System displays: Buildfile: /home/webtrust/Installs/lsaf-java-api-server-2.2/build.xml clean: clean-server1: do-clean: [echo] Clean previous Java API distribution in /sso/sfw/tcServer/lsafserver1/webapps/lsaf [delete] Deleting /sso/sfw/tcServer/lsafserver1/webapps/lsaf/WEB-INF/lib/sas.lsaf.api.client.jar [delete] Deleting /sfw/tcServer/lsafserver1/webapps/lsaf/WEB-INF/lib/sas.lsaf.api.server.jar clean-server2: clean-server3: clean-server4: ***** BUILD SUCCESSFUL Total time: 0 seconds	

Checklist A: "Install LSAF Java API"		Corresponding Document: <i>None</i>	
#	Activity	Expected Results	Actual Results
A18.	Type ant deploy	<p>System displays:</p> <p>Buildfile: /home/webtrust/Installs/lsaf-java-api-server-2.2/build.xml</p> <p>deploy: [echo] User home = /home/webtrust</p> <p>deploy-server1:</p> <p>do-deploy: [echo] Deploying distribution to = /sso/sfw/tcServer/lsafserver1/webapps/lsaf [copy] Copying 2 files to /sso/sfw/tcServer/lsafserver1/webapps/lsaf/WEB-INF [copy] Copying /home/webtrust/Installs/lsaf-java-api-server-2.2/WEB-INF/lib/sas.lsaf.api.client.jar to /sso/sfw/tcServer/lsafserver1/webapps/lsaf/WEB-INF/lib/sas.lsaf.api.client.jar [copy] Copying /home/webtrust/Installs/lsaf-java-api-server-2.2/WEB-INF/lib/sas.lsaf.api.server.jar to /sso/sfw/tcServer/lsafserver1/webapps/lsaf/WEB-INF/lib/sas.lsaf.api.server.jar</p> <p>deploy-server2:</p> <p>deploy-server3:</p> <p>deploy-server4:</p> <p>deploy-server5:</p> <p>deploy-server6:</p> <p>deploy-server7:</p> <p>deploy-server8:</p> <p>BUILD SUCCESSFUL Total time: 0 seconds</p>	
A19.	Type cd /sso/sfw/tcServer	Navigate to that directory.	

Checklist A: "Install LSAF Java API"		Corresponding Document: None	
#	Activity	Expected Results	Actual Results
A20.	Type <code>./tcruntime-ctl.sh lsafserver1 start</code>	<p>System displays:</p> <p>Using CATALINA_BASE: /sso/sfw/tcServer/lsafserver1 Using CATALINA_HOME: /sso/sfw/tcServer/tomcat-8.5.23.A.RELEASE Using CATALINA_TMPDIR: /sso/sfw/tcServer/lsafserver1/temp Using JRE_HOME: /sso/sfw/java/jdk1.8.0_92 Using CLASSPATH: /sso/sfw/tcServer/tomcat-8.5.23.A.RELEASE/bin/bootstrap.jar:/sso/sfw/tcServer/lsafserver1/bin/tomcat-juli.jar Using CATALINA_PID: /sso/sfw/tcServer/lsafserver1/logs/tcserver.pid Tomcat started. Status: RUNNING as PID=31186</p>	
A21.	Type <code>ps -ef grep webtrust</code>	<p>System displays:</p> <pre>webtrust 10208 1 0 Mar18 ? 02:15:01 /sso/sfw/java/jdk1.8.0_92/bin/java -Dnop - Djava.util.logging.manager=com.springsource.tcserver.servi ceability.logging.TcServerLogManager -Xss256K - Xms8192M -Xmx8192M -XX:NewSize=2048m - XX:MaxNewSize=2048m -XX:+UseConcMarkSweepGC - XX:+UseParNewGC -XX:+CMSScavengeBeforeRemark - XX:+CMSParallelRemarkEnabled -XX:+DisableExplicitGC -XX:+UseCompressedOops -XX:+UseLargePages - XX:LargePageSizeInBytes=2m -XX:+PrintGCDetails - XX:+PrintGCDateStamps -XX:+PrintFlagsFinal - Xloggc:/sso/sfw/tcServer/lsafserver1/logs/gc_hlsrd27au1_03 _18_2019_1041.log - XX:+HeapDumpOnOutOfMemoryError - XX:HeapDumpPath=/sso/sfw/tcServer/lsafserver1/logs - Djava.awt.headless=true - Djdk.tls.ephemeralDHKeySize=2048 - Djava.protocol.handler.pkgs=org.apache.catalina.webresourc es -classpath /sso/sfw/tcServer/tomcat- 8.5.23.A.RELEASE/bin/bootstrap.jar:/sso/sfw/tcServer/tomc at-8.5.23.A.RELEASE/bin/tomcat-juli.jar - Dcatalina.base=/sso/sfw/tcServer/lsafserver1 - Dcatalina.home=/sso/sfw/tcServer/tomcat- 8.5.23.A.RELEASE - Djava.io.tmpdir=/sso/sfw/tcServer/lsafserver1/temp org.apache.catalina.startup.Bootstrap start</pre>	
A22.	Type <code>cd lsafserver1/logs</code>	<p>Navigate to the /sso/sfw/tcServer/lsafserver1/logs directory.</p>	

Checklist A: "Install LSAF Java API"		Corresponding Document: <i>None</i>	
#	Activity	Expected Results	Actual Results
A23.	Type tail -f lsaf_info.log <i>Note:</i> Depending on how fast your server is, you may need to type tail -300 lsaf_info.log	You will see the contents of the log file while the server is starting up. Look for the following lines, indicating the LSAF API is installed and the success start of process. 2019-10-10 14:13:15,157 INFO LS_IN SAS Life Sciences Analytics Framework Java API is installed 2019-10-10 14:13:15,157 INFO LS_IN JAPI client version: 2.2.API.109.20191008.122537 2019-10-10 14:13:15,157 INFO LS_IN JAPI server version: 2.2.API.109.20191008.122537 2019-10-10 14:13:15,157 INFO LS_IN	
A24.	Hit Ctrl-C to stop viewing this log file.	You no longer see the output of the log file.	
A25.	Repeat steps A19 – A24 to start up any additional lsafservers. <i>Note:</i> In a clustered web server environment, steps A13 – A24 must be performed on all web servers in the cluster.	Steps repeated as necessary.	
A26.	Hit Ctrl-D to stop the script command.	System indicates script stopped running.	
A27.	Type exit	The webtrust userid is logged off the web server.	

Signature below indicates completion of Checklist A, items A1 – A27, above.

Name (print or type): _____

Sign-off : _____

Date: _____