



# SAS Life Science Analytics Framework Java API 2.1

---

Installation Qualification / Operational Qualification Checklist

Install LSAF Java API 2.1

**Version 1.00**  
**Date March 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 <b>ssh</b> connection from the admin PC to the <b>web server</b> and login as <b>webtrust</b> .	You're prompted for the password.	
A3.	Enter the <i>password</i> for webtrust.	You're logged into web server.	
A4.	Use the <b>script</b> command to capture commands that are typed in the Unix window. The syntax is <b>script /var/staging_logs/&lt;servername&gt;_lsafapi2.1_&lt;date&gt;.txt</b>	System should indicate script has started.	
A5.	From the admin PC, download the <b>LSAF Java API 2.1</b> distribution from the SAS Technical Support site. After the distribution has been downloaded, use the WinSCP2 tool to copy the file to <b>/home/webtrust/Installs</b> on the web server. Transfer it as a <b>binary</b> file using the <b>webtrust</b> userid.  <b>lsaf-java-api-server-2.1.zip</b>  <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 <b>cd /sso/sfw/tcServer</b>	Navigate to that directory.	

Checklist A: "Install LSAF Java API"		Corresponding Document: <i>None</i>	
#	Activity	Expected Results	Actual Results
A7.	Type <b>./tcruntime-ctl.sh lsafserver1 stop</b>	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 <b>ps -ef   grep webtrust</b>	You should not see the tcserver process running.	
A9.	Type <b>cd lsafserver1/logs</b>	Navigate to the /sso/sfw/tcServer/lsafserver1/logs directory.	
A10.	Delete or rename the <b>lsaf_info.log</b> and <b>catalina.out</b> files.	Log files are deleted or renamed.	
A11.	Type <b>ls -al</b>	Verify that log files have been renamed or deleted.	
A12.	Repeat <b>steps A6 – A11</b> for the remaining tcServers.  <i>Note:</i> You do not have to stop the <b>lsafserveradmin</b> 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 lsafserveradmin server.	
A13.	Type <b>cd /home/webtrust/Installs</b>	Navigate to that directory.	
A14.	Type <b>unzip lsaf-java-api-server-2.1.zip</b>	The contents of the zip file are extracted.	
A15.	Type <b>ls -al</b>	You will see a lsaf-java-api-server-2.1 directory that was created in the previous step.	
A16.	Type <b>cd lsaf-java-api-server-2.1</b>	Navigate to that directory.	

Checklist A: "Install LSAF Java API"		Corresponding Document: <i>None</i>	
#	Activity	Expected Results	Actual Results
A17.	Type <b>ant clean</b>	<p>System displays:</p> <p>Buildfile: /home/webtrust/Installs/lsaf-java-api-server-2.1/build.xml</p> <p>clean:</p> <p>clean-server1:</p> <p>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</p> <p>clean-server2:</p> <p>clean-server3:</p> <p>clean-server4:  *****</p> <p><b>BUILD SUCCESSFUL</b>  Total time: 0 seconds</p>	

Checklist A: "Install LSAF Java API"		Corresponding Document: <i>None</i>	
#	Activity	Expected Results	Actual Results
A18.	Type <b>ant deploy</b>	<p>System displays:</p> <p>Buildfile: /home/webtrust/Installs/lsaf-java-api-server-2.1/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-2.1/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-2.1/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 <b>cd /sso/sfw/tcServer</b>	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 <b>tail -f lsaf_info.log</b>  <i>Note:</i> Depending on how fast your server is, you may need to type <b>tail -300 lsaf_info.log</b>	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.  2019-04-10 11:44:28,558   INFO   LS_IN       SAS Life Sciences Analytics Framework Java API is installed 2019-04-10 11:44:28,559   INFO   LS_IN       JAPI client version: 2.1.API.206.20190410.112641 2019-04-10 11:44:28,559   INFO   LS_IN       JAPI server version: 2.1.API.206.20190410.112641 2019-04-10 11:44:29,249   INFO   LS_IN       Will start Quartz Scheduler [SAS_Scheduler] in 10 seconds 2019-04-10 11:44:39,253   INFO   LS_IN       Starting Quartz Scheduler now, after delay of 10 seconds	
A24.	Hit <b>Ctrl-C</b> to stop viewing this log file.	You no longer see the output of the log file.	
A25.	Repeat <b>steps A19 – A24</b> 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 <b>Ctrl-D</b> to stop the script command.	System indicates script stopped running.	
A27.	Type <b>exit</b>	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: \_\_\_\_\_