General Availability 3.0 Release

FY2015

INSTALLATION ORACLE WEBLOGIC APP FOR SPLUNK

FUNCTION1

Oracle WebLogic App for Splunk (v. 3.0 GA)

Introduction

Thank you for downloading the 3.0 GA release of Oracle WebLogic App for Splunk, by Function1. Function1 (www.function1.com) is a leader in the Operations Intelligence and Middleware space. We have designed the base architecture for some of the largest Splunk deployments in the world today, and have helped develop the standard for enterprise class governance and data on-boarding. Our strategic relationship with Splunk has provided a unique perspective that empowers our Professional Services Organization to bridge the gap between application and infrastructure management. Function1 team members are not only technology savvy, but are also business professionals, and able to provide the full picture, outlook, and strategy for your organization. Function1 is proud to serve multiple industries and delivers broad industry experience to your deployment. Our combined talent and years of experience in enterprise software deployments will help you gain more value from your technology investment.

We are committed to the success of your deployment and would be happy to assist. Contact support@function1.com for assistance.

Technology Add-on Requirements

The Oracle WebLogic App for Splunk and the associated Technology Add-ons (TA) are designed for installation on a Splunk Universal Forwarder on WebLogic (WLS) components. These TAs poll your WLS AdminServers by the minute, hour, and daily for JMX data, and outputs the generated logging into monitored log files at /$SPLUNK_HOME/var/log. They monitor OS performance data, as well as gather data from Managed and Unmanaged nodes.

The app itself, inclusive of the TAs, is highly dynamic as well as independent to maximize on the native behavior and functionality within WebLogic as well as Splunk. The TAs include all the necessary technology to bridge the gap between these two technologies.

Product documentation is widely available, though we highly recommend planning your deployment with a member of our Professional Services team. Please contact support@function1.com for more assistance.

Before Getting Started

Since the TAs rely heavily on the interaction between a Splunk forwarder and WLS, the service accounts that run these applications will require access to each other's file paths. While both applications are often run by local system in a Windows environment, in a Unix environment these applications are more often run by separate service accounts. In this event, the Splunk and WebLogic service accounts will require full access to each other's respective application file paths.

Deploy the Oracle WebLogic App for Splunk

There are only a couple of steps you will need to take to successfully install the Oracle WebLogic App for Splunk. First, you will need to install all of the TAs to their proper locations and perform some slight configuration changes. Then, you will need to configure your core Splunk instance to properly receive, index, and present the WLS data.

Install the Technology Add-ons and the App

1. Download the Oracle WebLogic App for Splunk from Splunkbase and unpack it to an accessible location.
2. Download and install Splunk universal forwarders (UF) to each WebLogic component. Keep in mind that you only need to install one UF per server, even if the server houses multiple WebLogic roles.
3. Deploy the TAs to your WLS environment.
    a. The TAs are located in Function1_WebLogicServer/appserver/addons
    b. If you are using the Splunk deployment server, place the TAs in $SPLUNK_HOME/etc/deployment-apps on the deployment server. You must also configure serverclass.conf on the deployment server. For more information about distributed environments and how to manage and configure them look here: http://docs.splunk.com/Documentation/Splunk/latest/Deploy/Distributedoverview
    c. Review the table below and deploy, or install, the TAs to their corresponding role(s):

Server role:	OS	Deploy these TA(s)
WLS AdminServer	Windows	Function1_WLS_AdminServer_Win_TA Function1_WLS_Win_Perf_TA*
	*Nix	Function1_WLS_AdminServer_Nix_TA Function1_WLS_Nix_Perf_TA*
WLS ManagedServer and/or WLS NodeManager	Windows	Function1_WLS_ManagedServer_Win_TA Function1_WLS_Win_Perf_TA*
	*Nix	Function1_WLS_ManagedServer_Nix_TA Function1_WLS_Nix_Perf_TA*

* If you are already using the Splunk for Windows technology add-on, and/or the Splunk for Unix and Linux apps to monitor OS performance, you should not install the Function1_WLS_Win_Perf_TA or Function1_WLS_Nix_Perf_TA. The TAs will index the same data as the Splunk for Windows technology add-on and the Splunk for Unix and Linux app.

    d. We highly recommend using a deployment server to distribute the TAs, but if you are not, then manually install them on each UF in $SPLUNK_HOME/etc/apps.
    e. Once the UFs have their appropriate TAs, you must configure your core Splunk instance. Review the table below:

Core Splunk Instance	Install on Indexer	Install on Search Head
Single server: Search Head and Indexer combination	Function1_WLS_Core_TA Function1_WebLogicServer	N/A

Distributed environment: Multiple Search heads and/or multiple Indexers	Function1_WLS_Core_TA	Function1_WLS_Core_TA Function1_WebLogicServer

f. Now your core Splunk instance will be able to receive, index, parse, and display the data from your WLS environment.

Configure the AdminServer TAs

Configure the Function1_WLS_AdminsServer* TAs to allow the scripts within the TA to run successfully.

For *Nix environments

1. Within the Function1_AdminServer_Nix_TA, copy the inputs.conf file in the ../default directory to the ../local directory. Note: if your WLS hosts have different installation paths for WebLogic Server, then you will need to create a TA for each different WLS configuration. You can rename the TA accordingly.
2. In the ../local/inputs.conf file
   a. Under '*Nix JMX Input Scripts', replace '/your/app/file path' with the full path to the TA, remember to account for changes you may have made to the name of the TA. Do this for all three scripted inputs: "EVERY MINUTE," "EVERY HOUR," and "EVERY DAY"
   b. Replace '/your/weblogic/home' with the full file path to your WebLogic home. Repeat for all three scripted inputs. When complete, your scripted input stanzas should look like the example below:

```
# RUN PY TO WLST TO MBEAN AND WRITE JMX LOG
### *Nix JMX Input Scripts
# EVERY MINUTE
[script://./bin/runWlstScriptsMinute.sh /opt/splunkforwarder/etc/apps/function1_weblogicserver_ta_nix /opt/weblogic]
disabled = false
index = wls
sourcetype = wls_trash
interval = 300

# EVERY HOUR
[script://./bin/runWlstScriptsHourly.sh /opt/splunkforwarder/etc/apps/function1_weblogicserver_ta_nix /opt/weblogic]
disabled = false
index = wls
sourcetype = wls_trash
interval = 3600

# EVERY DAY
[script://./bin/runWlstScriptsDaily.sh /opt/splunkforwarder/etc/apps/function1_weblogicserver_ta_nix /opt/weblogic]
disabled = false
index = wls
sourcetype = wls_trash
interval = 86400

# FORWARD JMX LOG
[monitor://$SPLUNK_HOME/var/log/wls_jmx*]
disabled = false
index = wls
sourcetype = wls_jmx
```

Important Note: When configuring your local/inputs.conf file, be sure to remember to replace '/your/app/filepath' and '/your/WebLogic home' with your actual file paths in all scripted input stanzas**.**

3. Change directory to .../bin in the TA. In "setWlstEnv.sh" set your WebLogic domain, AdminServer name, and admin port.
    a. DOMAIN_COUNT should reflect the number of domains you declare in this file.
    b. "ADMIN_SERVER" should be set to the name of your WLS Admin Server as configured in the WLS config.xml file for this WLS domain.
    c. "ADMIN_PORT" should be set to the port the WLS Admin Server is listening on as configured in the WLS config.xml file for this WLS domain. Your configuration should looks similar to below

```
##Set your total number of domains here
export DOMAIN_COUNT=1

## For each domain on this server, add the domain path, adminServerName and admin port
## Please make sure that the entries are in the respective order
export DOMAIN_PATH_1=/opt/weblogic/user_projects/domains/wcsDomain/
export ADMIN_SERVER_1=AdminServer
export ADMIN_PORT_1=7001
```

4. In the "wlsCollectDataDaily.py," "wlsCollectDataHourly.py", and "wlsCollectDataMinute.py" files make sure "admin_url" is set to the "listen-address" as configured in the WLS config.xml file for this WLS domain. The following is an example of the AdminServer section in a WLS config.xml file

```
<server>
    <name>AdminServer</name>
  <machine>host123.acme.com</machine>
  <listen-port>7001</listen-port>
  <listen-address>host123.acme.com</listen-address>
  <server-diagnostic-config>
   <name>AdminServer</name>
   <diagnostic-context-enabled>true</diagnostic-context-enabled>
  </server-diagnostic-config>
 </server>
```

In the "wlsCollectData*.py" files, the "admin_url" would be set as the following:

```
admin_url = "t3://host123.acme.com:" + adminsvrport
```

Note: If no listen-address is specified for the AdminServer in the WLS config.xml file, then "admin_url" can be set to "localhost"

5. Once you have completed these configurations, deploy the TA to the forwarder residing on the WLS Admin Server by either using the Splunk deployment server or copying the TA to the forwarder manually. If you manually copy the TA to the forwarder, be sure to restart the forwarder once your

updates are complete.

For Windows environments

1. Within the Function1_WLS_AdminServer_Win_TA, use a text editor to open the file \default\inputs.conf. Under 'Windows JMX Input Scripts', replace 'C:\your\app\filepath' with the full path to your app (including the name of your app). Do this for all three scripted inputs.
2. Replace 'C:\your\weblogic\home' with the full file path to your WebLogic home. Repeat for all three scripted inputs.
3. Once complete, go to …<yourTA>\local\. Create a copy of inputs.conf.enablewin called simply inputs.conf. Then, create a copy of perfmon.conf.enablewin called simply perfmon.conf. These files will allow you to toggle the various input monitoring features of this TA on and off by changing "disabled = false" to "disabled = true" in any stanza. **I**
    - Important **Note**: When configuring your local\inputs.conf file, be sure to remember to replace 'C:\your\app\filepath' and 'C:\your\weblogic\home' with your actual file paths in all scripted input stanzas.
4. Once you have enabled/disabled your desired features, go to <yourTA>\bin. Using an editor, open setWlstEnv.cmd. Here you will set your WebLogic domain, AdminServer name, and admin port. Your DOMAIN_COUNT should reflect the number of domains you declare in this file. Example:

```
@rem ## For each domain on this server, add the domain path, adminServerName and admin port
@rem ## Please make sure that the entries are in the respective order
set DOMAIN_PATH_1=C:\oracle\Middleware\wlserver_10.3\samples\domains\medrec
set ADMIN_SERVER_1=MedRecServer
set ADMIN_PORT_1=7011
```

5. Once you have completed these configurations, restart your forwarder to allow the changes to take effect. Verify data is generating by checking the log files under $SPLUNK_HOME\var\log\.

Troubleshooting the AdminServer Technology Add-on

In order to test whether or not the AdminServer TA is sending the correct data to Splunk, you can run the following search in Splunk:

index=wls sourcetype=wls_jmx* host=<name_of_host>

<name_of_host> = the name of the WLS host you are expecting data from. You can also leave out this field if wanting to see data from all WLS hosts.

Here are some steps to help troubleshoot the installation of the technology add-on:

1. To verify that the scripts are working, navigate to the **$SPLUNK_HOME/var/log/** directory on the server the TA is being deployed to and verify that three files exist: "wls_jmx_daily.log" "wls_jmx_hourly.log" and "wls_jmx_minute.log"
    a. If the files do not exist then there is an issue with the scripts being run. Proceed to Step 2.

b. If those files exist and have data in them, then there is an issue with the TA picking up the files and forwarding them. Proceed to Step 3.

2. Navigate to the AdminServer TA's bin directory in $SPLUNK_HOME/etc/apps/Function1_WLS_AdminServer_<Nix or Win>_TA/bin on the forwarder. And run any of the "runWlstScripts*.sh" (Unix/Linux environment) or "runWlstScripts*.cmd" (Windows environment) files from the command-line. Examine the output to determine if any errors occurred. Note, that the WLST script which is invoked by the shell script will try to output a log of the script execution to a directory under the WLS installation path, however, unless the account Splunk is running under has write access to this directory it will throw an error, so any errors related to this can be considered "normal" and ignored. Resolve any other errors that the script generates. Go to Step 1.

3. Verify the TA's inputs.conf file is correct. Navigate to the $SPLUNK_HOME/etc/apps/ Function1_WLS_AdminServer_<Nix or Win>_TA/local directory. In the inputs.conf file, ensure that the [monitor://$SPLUNK_HOME/var/log/wls_jmx_*] stanza is enabled.
    a. If it is not enabled, then enabled it on the deployment server and re-deploy the TA to the forwarder.
    b. If it is enabled, then go to Step 4

4. Verify the forwarders output configuration. Navigate to $SPLUNK_HOME/etc/app/<name_of_outputs_package>/local directory. Verify the indexer's settings are correct in the "outputs.conf" file.
    a. If the "outputs.conf" file is incorrect or the package is missing then correct on the deployment server and re-deploy.
    b. If the outputs package appears to be correct and the forwarder is sending other data to Splunk, then the issue should be escalated.

TA Inventory and Configuration Options

This section will review each TA and the configuration files that are within them, and will provide you with the information to customize them.

AdminServer TAs:

- Scripts in the /bin directory provide JMX data and set environment variables.
- inputs.conf – all inputs are enabled by default.
    - To disable inputs, change "disabled = false" to "disabled = true"
- Data is indexed in the "wls" index, which is native to the Oracle WebLogic App for Splunk.

ManagedServer TAs:

- inputs.conf – all inputs are enabled by default.
    - To disable inputs, change "disabled = false" to "disabled = true"
- Node Manager logs are also being monitored.

    - To disable inputs, change “disabled = false” to “disabled = true”
- Data is indexed in the “wls” index, which is native to the Oracle WebLogic App for Splunk.

Windows Perf TA:

- perfmon.conf – all inputs are enabled by default.
    - The default interval has been changed from 10 seconds, as it is in the Splunk for Windows technology add-on, to 60 seconds.
    - The following stanza as been added to capture process data:

```
[PERFMON:CPUTime_process]
counters = % Processor Time;% User Time
disabled = false
instances = *
interval = 60
object = Process
```

    - If you have the Splunk for Windows technology add-on installed, be sure to add the previous stanza to perfmon.conf. If you do not, the Oracle WebLogic App for Splunk will not perform properly.
    - If you are using Splunk 5+, disable all stanzas in this file and ensure to enable all stanzas in inputs.conf.
    - The Oracle WebLogic App for Splunk only uses perfmon data for its views. If you are using the Splunk for Windows technology add-on you must enable perfmon input stanzas or update the views to use WMI data.
    - To disable inputs, change “disabled = false” to “disabled = true”
- wmi.conf – all inputs are disabled by default.
    - The Oracle WebLogic App for Splunk only uses perfmon data for its views. If you are using the Splunk for Windows technology add-on you must enable perfmon input stanzas or update the views to use WMI data.
    - To enable inputs, change “disabled = true” to “disabled = false”
- inputs.conf – all inputs are disabled and commented out by default.
    - If you are using Splunk 5+, uncomment the stanzas, enable all inputs and disable all stanzas in perfmon.conf.
    - To enable inputs, change “disabled = true” to “disabled = false”
    - NOTE: If you are not running Splunk 5+, and the stanzas are disabled but are not commented out, it will produce syntax errors during start up.
- Data is indexed in the “main” index, which is the default index for Windows performance data.

****Nix Perf TA***:

- Scripts in the /bin directory provide *nix performance data.
- inputs.conf – all inputs are enabled by default.
    - To disable inputs, change “disabled = false” to “disabled = true”
- Data is indexed in the “os” index, which is the default index for *Nix performance data.

Core TA:

- indexes.conf – creates the “wls” and “os” indexes
    - Indexes are set to go to the default $SPLUNK_DB directory. If you are using a different path, copy the indexes.conf in /default and place it in /local and make your updates there.
- props.conf – includes all index time and search time props.
    - To make changes to this configuration file, copy the version in /default and place it in /local and make all updates to that version only.
- transforms.conf – includes all tranformations necessary for the indexer(s) and the search head(s) to index and interpret the WLS data.
    - To make changes to this configuration file, copy the version in /default and place it in /local and make all updates to that version only.
- eventtypes.conf – knowledge objects used to enhance the Oracle WebLogic App for Splunk app.
    - To make changes to this configuration file, copy the version in /default and place it in /local and make all updates to that version only.
- tags.conf – knowledge objects used to enhance the Oracle WebLogic App for Splunk app.
    - To make changes to this configuration file, copy the version in /default and place it in /local and make all updates to that version only.

Troubleshooting, Support, & Feature Enhancements

We welcome the opportunity to work with you and help overcome any hurdle in your path to a successful deployment. The Oracle WebLogic App for Splunk has passed three private beta releases and the Public Beta has been available for a few months and has received positive reviews. This General Availability 1.0 is designed to ensure that the most commonly desired features are included. If you would like custom functionality, additional features, help deploying, or wish to report a bug, we would love to hear from you. Please contact support@function1.com for assistance.

Known Issues

SPLKWLS-84	Topology Page - "Servers in Domain" Panel - Consistency with Machines List
SPLKWLS-83	Topology Page - Machines in Domain Panel - Consistency with Machines List
SPLKWLS-85	Applications Page - WLS Server List is Incomplete
SPLKWLS-121	Server Overview Page - Does Not Load Correctly When Navigating from Applications Page