



Purpose

This document will cover how to monitor WebLogic servers using the WebLogic wizard with Nagios XI, so that users may be notified when WebLogic applications are behaving unexpectedly.

Target Audience

This document is intended for use by Nagios XI Administrators who want to monitor their WebLogic instances.

Prerequisites

This document assumes you have the following:

- A remote WebLogic server
- A Nagios XI server with a network route to the WebLogic server

The monitoring of the WebLogic server is performed by the [WLSAgent](#), this can be accessed via HTTP calls from the Nagios XI server or via [NCPA](#). Either method requires some prerequisite steps to be followed first which are outlined below.

WLSAgent

The WLSAgent is a Java application that allows you to query the WebLogic server. You will need to download the WLSAgent to a location on the WebLogic server, it can be downloaded directly from the Nagios XI server. This example is downloading it to the `oracle` user directory `/home/oracle` and may be different in your environment.

In our example the linux user that runs WebLogic is `oracle` and you will need to execute all following commands as this user.

In the following commands replace `xi_address` with the IP address of your Nagios XI server. In a terminal

session on the WebLogic server and execute the following commands:

```
cd /home/oracle
wget http://xi_address/nagiosxi/includes/configwizards/java-as/plugins/wlsagent.tar.gz
tar xvf wlsagent.tar.gz
```

The WLSAgent will listen on port 9090 by default, if you want to change this edit the `/home/oracle/wlsagent/run.sh` file and change the `PORT` line.

You now need to generate and copy the WebLogic client library files. This example using using WebLogic that is installed in `/home/oracle/wls12213` however your location may differ. This has been set to the environment variable `ORACLE_HOME` as per the following command:

```
export ORACLE_HOME=/home/oracle/wls12213
```

Execute the following commands to generate and copy the WebLogic client library files:

```
cd $ORACLE_HOME/wlserver/server/lib
java -jar ../../modules/com.bea.core.jarbuilder.jar
cp ./{wlclient.jar,wljmxclient.jar,wlfullclient.jar} /home/oracle/wlsagent/lib
```

You are now ready to start the WLSAgent, execute the following command to do so:

```
/home/oracle/wlsagent/run.sh
```

This should execute silently and return to the command prompt. Execute the following command to determine if it is running:

```
ps -ef | grep wls
```

This should output something like:

```
oracle      3115          1   1 10:48 pts/1      00:00:00 /bin/java -Xmx32m -cp
.:wlsagent.jar:lib/servlet-api-2.5.jar:lib/jetty-servlet-
7.6.2.v20120308.jar:lib/wlclient.jar:lib/wljmxclient.jar:lib/wlfullclient.jar
net.wait4it.nagios.wlsagent.core.WLSAgent 0.0.0.0 9090
```

This indicates that the WLSAgent application is running.

You may also need to open the firewall port 9090, depending if your operating system is using a firewall (*step not required if using NCPA*). In our example the WebLogic server is running on CentOS 7 and the following commands are required (you will need to execute these as a root user):

```
firewall-cmd --zone=public --add-port=9090/tcp
firewall-cmd --zone=public --add-port=9090/tcp --permanent
```

Once these steps have been performed the WLSAgent is installed and ready to be used. If you are connecting to the WLSAgent from the Nagios XI server using HTTP calls please proceed to the [Configuration Wizard](#) section of this document.

If you do not wish to connect to the WLSAgent from the Nagios XI server using HTTP calls then you will need to install and configure [NCPA](#) to allow you to communicate with the WLSAgent.

NCPA

If the WLSAgent cannot be used then NCPA will need to be installed on the WebLogic server as per the [Installing NCPA](#) documentation.

Once installed you will also need to copy the `check_wlsagent.sh` plugin from the Nagios XI server to your

WebLogic server. The following commands executed on your WebLogic server as the root user will do this:

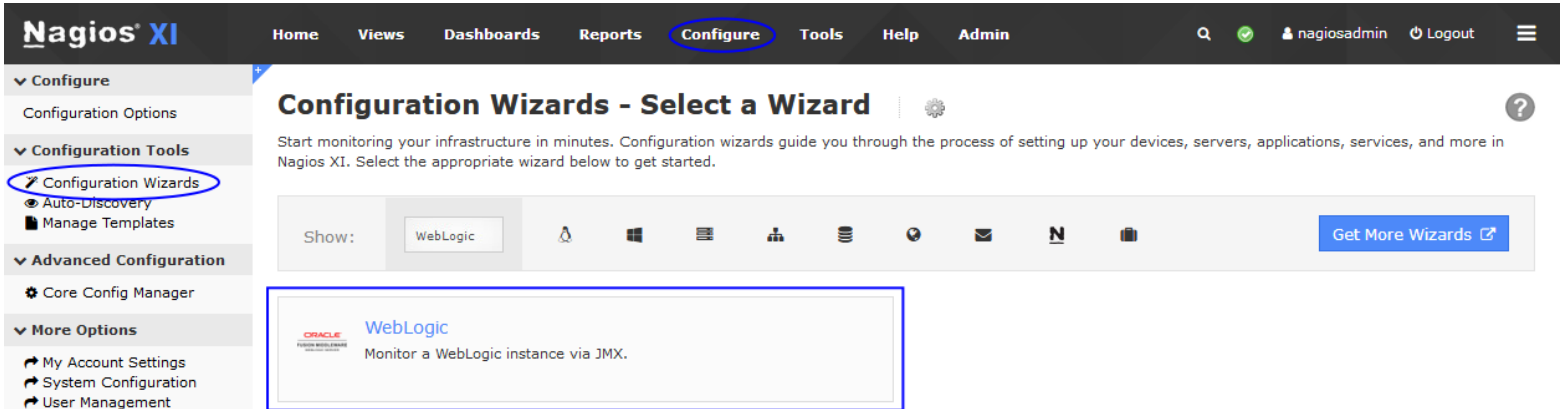
```
cd /usr/local/ncpa/plugins
wget http://xi_address/nagiosxi/includes/configwizards/java-as/plugins/check_wlsagent.sh
chmod +x check_wlsagent.sh
chown nagios:nagios check_wlsagent.sh
```

Once you've completed these steps please proceed to the [Configuration Wizard](#) section of this document.

The WebLogic Configuration Wizard

The WebLogic config wizard uses the WLS Agent to retrieve system statistics and compares them to the thresholds you set in the wizard. Checks can either be combined into one service or separated.

To begin using the WebLogic configuration wizard, navigate via the top bar to **Configure > Configuration Wizards**. Then, select the **WebLogic** wizard. In the following screenshot you can see how the search field allows you to quickly find a wizard.



The screenshot shows the Nagios XI interface. The top navigation bar includes 'Home', 'Views', 'Dashboards', 'Reports', 'Configure', 'Tools', 'Help', and 'Admin'. The 'Configure' menu is expanded, showing 'Configuration Options', 'Configuration Tools', 'Advanced Configuration', and 'More Options'. Under 'Configuration Tools', 'Configuration Wizards' is selected and circled in blue. Below this, a search field contains 'WebLogic', and the 'WebLogic' wizard is highlighted with a blue box. The wizard description reads: 'Monitor a WebLogic instance via JMX.'

Step 1 requires you to provide the details for the Nagios XI server to connect to WebLogic server.

There are two methods of connecting to WebLogic, **Direct HTTP connection** or **Remote Agent (NCPA)**.

Both methods require the same settings however NCPA has some additional options.

An explanation of the fields you are required to provide are explained as follows.

- **Nagios Hostname** is the name that you want the monitoring objects created by the wizard to be associated with in Nagios XI
- **DNS Hostname/IP Address** is the network address of the WebLogic server
- **WLSAgent Port** is how Nagios XI or NCPA connects to the WLSAgent
- **JMX T3 Port** is how the WLSAgent connects to WebLogic. Depending on the version of WebLogic this can be 7001 or 8001. You can see this in the WebLogic logs:
 - `<Sep 12, 2018 10:42:26,239 AM AEST> <Notice> <Server> <BEA-002613> <Channel "Default" is now listening on 10.25.9.4:7001 for protocols iiop, t3, ldap, snmp, http.>`
- **WebLogic Username** and **WebLogic Password** are the credentials required to access the WebLogic internal statistics
- **NCPA Listener Port** and **NCPA Token** only appear when you select the NCPA access method. You defined these options when installing NCPA earlier.

Configuration Wizard: WebLogic - Step 1

Plugin and Agent Setup

You will need to set up WLSAgent on your WebLogic server in order to monitor it. You may or may not also need NCPA installed on the server, depending on your monitoring preferences.

- [Download the WLS Agent](#)
- [Additional documentation for NCPA](#) and for [monitoring application servers](#).

WebLogic Server Information

Access WebLogic via:

Nagios Hostname

The host name you want associated with this check.

DNS Hostname/IP Address

The hostname/IP address of your WebLogic instance.

WLSAgent Port

JMX T3 Port

WebLogic Username

The username for the WebLogic domain to monitor.

WebLogic Password

The password for the WebLogic user above.

NCPA Information

NCPA Listener Port

NCPA Token

After making all your selections click **Next** to proceed to **Step 2**.

Step 2 provides you with multiple monitoring options.

In **Service Information** you can define the **Service Description** of the service that will be created by this wizard. By default the checks below are combined into one service, de-selecting the check box will create multiple services (the name of each service created will begin with the defined service description).

Each metric allows you to check and un-check the relevant boxes to determine which checks to run, and enter your desired warning and critical thresholds.

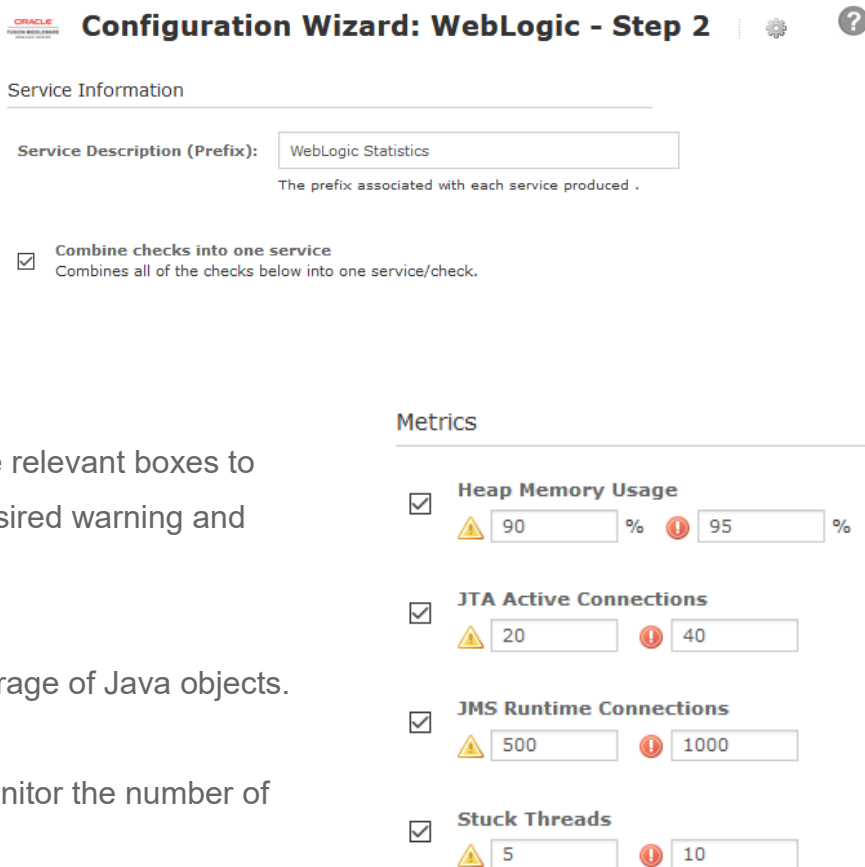
Heap Memory is the memory segment for the storage of Java objects.

JTA is the Java Transaction API, here you can monitor the number of active connections.

JMS is the Java Message Service, here you can monitor its overall number of connections.

Stuck Threads in WebLogic can also be monitored.

The remaining checks require you to define a name (Datasource / Application / Queue) and the relative thresholds.



Configuration Wizard: WebLogic - Step 2

Service Information

Service Description (Prefix):
The prefix associated with each service produced .

Combine checks into one service
Combines all of the checks below into one service/check.

Metrics

Heap Memory Usage
Warning: % Critical: %

JTA Active Connections
Warning: Critical:

JMS Runtime Connections
Warning: Critical:

Stuck Threads
Warning: Critical:

If you don't want to define a name then use the wildcard * to monitor all of them.

Click Next and then complete the wizard by choosing the required options in Step 3 – Step 5.

To finish up, click on **Finish** in the final step of the wizard.

Once the wizard applies the configuration, click the **View status details for <your host>** link to see the new service(s) that have been created.

Here is an example of all checks in one service:

Host	Service	Status	Duration	Attempt	Last Check	Status Information
WebLogic	WebLogic Statistics	Ok	2m 20s	1/5	2018-09-18 12:02:16	myserver is in RUNNING state, status OK

Here is an example of multiple checks in separate services:

Host	Service	Status	Duration	Attempt	Last Check	Status Information
WebLogic Server	WebLogic Statistics: component	Ok	5m 50s	1/5	2018-09-18 12:07:56	myserver is in RUNNING state, status OK
	WebLogic Statistics: jdbc	Ok	6m 12s	1/5	2018-09-18 12:07:34	myserver is in RUNNING state, status OK
	WebLogic Statistics: jms_queue	Ok	6m 17s	1/5	2018-09-18 12:07:29	myserver is in RUNNING state, status OK
	WebLogic Statistics: jms_runtime	Ok	6m 26s	1/5	2018-09-18 12:07:20	myserver is in RUNNING state, status OK
	WebLogic Statistics: jta	Ok	6m 8s	1/5	2018-09-18 12:07:38	myserver is in RUNNING state, status OK
	WebLogic Statistics: jvm	Ok	5m 54s	1/5	2018-09-18 12:07:52	myserver is in RUNNING state, status OK
	WebLogic Statistics: thread_pool	Ok	6m 3s	1/5	2018-09-18 12:07:43	myserver is in RUNNING state, status OK

JDBC Waiting Connections

Each name is the datasource's JNDI name. Enter the wildcard (*) to apply the same warning/critical thresholds to all datasources.

Datasource Name	Warning Threshold	Critical Threshold
<input type="text"/>	<input type="text"/>	<input type="text"/>

[Add Row](#) | [Delete Row](#)

Component HTTP Sessions

Each application name is its exact context root. Enter the wildcard (*) to set the same warning/critical thresholds for all applications.

Application Name	Warning Threshold	Critical Threshold
<input type="text"/>	<input type="text"/>	<input type="text"/>

[Add Row](#) | [Delete Row](#)

JMS Queue Message Count

Each queue name is a JMS resource's WebLogic Name. Enter the wildcard (*) to apply the same warning/critical thresholds to all datasources.

Queue Name	Warning Threshold	Critical Threshold
<input type="text"/>	<input type="text"/>	<input type="text"/>

Finishing Up

This completes the documentation on how to monitor WebLogic in Nagios XI.

If you have additional questions or other concerns, please visit us at our support forums:

<https://support.nagios.com/forum>

The Nagios Support Knowledgebase is also a great support resource:

<https://support.nagios.com/kb>