

Troubleshooting Guide and FAQs

Community 1.5.1 release

This guide details the common problems which might occur during Jumbune execution, help you to diagnose the problem, and provide probable solutions.

Table of Contents

General Troubleshooting	3
Problem Identification	3
Verify the system and environment requirements	3
Collecting information	3
Troubleshooting Tools	3
Common Issues	3
Deployment	3
JSON Validation	3
Module Issues	4
Frequently Asked Questions	5

General Troubleshooting

Problem Identification

As you start identifying the problem, follow the basic checklist:

Verify the system and environment requirements

System and environment requirements are one of the most overlooked causes for failure and can also lead to performance issues. Please take the requirements into consideration while deploying Jumbune. Further information on the requirements can be obtained from the [Installation Guide](#).

Collecting information

Collecting and reviewing necessary information is important to drill down closest to the issue/cause. You can collect information from the following places:

- Examining logs (Jumbune deployment and Jumbune Agent)
- Check application Environment variables, System Configuration.

Troubleshooting Tools

Several tools can be used to collect information for troubleshooting purpose. Some of the commonly used tools are:

- JConsole for JMX monitoring.
- Operating system commands for examining the system state.
- JHat for examining the thread and heap dumps.

Common Issues

Deployment

Lib directory is not getting created inside the \$JUMBUNE_HOME folder.

If the distribution jar is placed in the JUMBUNE_HOME directory and the deployment is triggered from the same \$JUMBUNE_HOME directory, the \$JUMBUNE_HOME/lib directory gets deleted. This is a known bug. To avoid this, please ensure the distributed jar is not placed and triggered from within the \$JUMBUNE_HOME directory.

JSON Validation

Response of the JSON form validation doesn't come after a long time.

These are the probable reason for not getting any response from server while validating JSON:

- Hadoop daemons are not started, check hadoop daemons by executing jps command.
- Agent is not running.
- Agent port is not specified correctly.

“Jar not present” exception traced in logs.

Probable reasons for this exception are as follows:

- Specified jar path for Jumbune job is incorrect.
- JSON form does not retain the uploaded Jumbune job jar path while reusing JSON. Browse each time when running Jumbune job from Local repository.

Module Issues

Debugger and Profiler are not running, all configurations seem to be appropriate and data validation and cluster monitor modules are running fine.

Probable reason for this issue is machine name to IP address mapping is missing or is in incorrect form. Perform the following steps to resolve this problem:

Check the logs

- If the log displays ‘**Class not found**’ exception, run the following command to ensure that a path separator is added at the end of the path:

```
echo $JUMBUNE_AGENT
```

- If the log displays ‘**Path already exist**’ exception, the output directory on the HDFS might already be present. Provide a non-existent path.

Debugger logs are not getting created at the JUMBUNE_HOME/jobJars</job_name>/logs directory.

Please check that you have installed “*zip utility*” in every node of the cluster.

Cluster monitoring not working, notifying a “Failed Job Error”.

Please clear browser cache, this is a known issue.

“Currently executing a job” error message display on to the UI without launching any job.

When you run Jumbune from multiple browsers, there is a possibility of multiple submitted JSON being successfully validated at the same time. In that case, multiple Jumbune jobs will attempt to run. Since only one profiling job can be executed at an instance, other uses will get the above message. Please retry running profiling job once the running job completes.

Frequently Asked Questions

How can I change the logging level of the Jumbune?

You can change it by modify the <JUMBUNE_HOME>/resources/log4j2.xml file. Default logging level is in INFO level.

How can I change the verbosity of Jumbune agent?

Add the following word at the end of Jumbune agent execution command:

-verbose

How can I contribute in Jumbune?

You may contribute to Jumbune in following ways:

- Joining mailing list
- Report bugs / fix bugs
- Become a contributor / committer
- Provide feedback and improvement suggestions

Please refer to [Getting Involved document](#) for more information.