

JobScheduler Universal Agent - Logging

- [Scope](#)
- [Log Rotation](#)
 - [Rotating the log depending on a period](#)
 - [Rotating the log depending on the size](#)
- [Log Rotation and Compression](#)
 - [Rotating and compressing the log depending on a period](#)
 - [Rotating and compressing the log depending on the size](#)

Scope

- For general information about JobScheduler Master and Agent logging see [What logging possibilities does JobScheduler provide?](#)
- This article is focused on log configuration settings that are applied in some typical scenarios, log rotation and compression.
- Detailed information on log configuration is available from [Apache Log4j](#).
- Modifications of the log configuration are applied with the configuration file ``${SCHEDULER_HOME}/lib/log4j.xml`

Log Rotation

Rotating the log depending on a period

Rotating the log depending on a period

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE log4j:configuration SYSTEM "log4j.dtd">
<log4j:configuration debug="false" xmlns:log4j="http://jakarta.apache.org/log4j/">
  <appender name="CONSOLE" class="org.apache.log4j.ConsoleAppender">
    <layout class="org.apache.log4j.PatternLayout">
      <param name="ConversionPattern" value="%d{yyyy-MM-dd HH:mm:ss,SSS Z} [%-5p] %c - %m%n" />
    </layout>
  </appender>
  <appender name="FILE" class="org.apache.log4j.DailyRollingFileAppender">
    <!-- not supported by DailyRollingFileAppender param name="MaxBackupIndex" value="7"/-->
    <param name="Append" value="true" />
    <param name="File" value="${LOGFILE}.log" />
    <param name="DatePattern" value="'. 'yyyy-MM-dd" />
    <layout class="org.apache.log4j.PatternLayout">
      <param name="ConversionPattern" value="%d{yyyy-MM-dd HH:mm:ss,SSS Z} [%-5p] %c - %m%n" />
    </layout>
  </appender>
</root>
  <priority value="debug"/>
  <appender-ref ref="FILE"/>
</root>
</log4j:configuration>
```

The entry `<param name="DatePattern" value="'. 'yyyy-MM-dd"/>` specifies the rotation frequency:

Frequency	Configuration Sample
Rotate every minute	<code><param name="DatePattern" value="'. 'yyyy-MM-dd-hhmm" /></code>
Rotate every hour	<code><param name="DatePattern" value="'. 'yyyy-MM-dd-hh" /></code>
Rotate every day	<code><param name="DatePattern" value="'. 'yyyy-MM-dd" /></code>

Rotating the log depending on the size

Rotating the log depending on the size

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE log4j:configuration SYSTEM "log4j.dtd">
<log4j:configuration debug="false" xmlns:log4j="http://jakarta.apache.org/log4j/">
  <appender name="CONSOLE" class="org.apache.log4j.ConsoleAppender">
    <layout class="org.apache.log4j.PatternLayout">
      <param name="ConversionPattern" value="%d{yyyy-MM-dd HH:mm:ss,SSS Z} [%-5p] %c - %m%n" />
    </layout>
  </appender>
  <appender name="FILE" class="org.apache.log4j.RollingFileAppender">
    <param name="Append" value="true" />
    <param name="File" value="${LOGFILE}.log" />
    <param name="MaxFileSize" value="2KB" />
    <param name="MaxBackupIndex" value="5" />

    <layout class="org.apache.log4j.PatternLayout">
      <param name="ConversionPattern" value="%d{yyyy-MM-dd HH:mm:ss,SSS Z} [%-5p] %c - %m%n"/>
    </layout>
  </appender>
</root>
  <priority value="debug"/>
  <appender-ref ref="FILE"/>
</root>
</log4j:configuration>
```

This will create a new file `.log.n` ($n=1..5$) when the size exceeds 2KB. You also can specify 20MB etc. The maximum value for n is 13.

Log Rotation and Compression

If you want to compress the rotated logs, please check the `$(SCHEDULER_HOME)/lib/log/log4j` folder for availability of the file `apache-log4j-extras-1.2.xx.jar`.

It should be provided with [SET-62 - Getting issue details...](#)

Should this file not be available then download it from <https://logging.apache.org/log4j/extras/download.html>

Rotating and compressing the log depending on a period

Rotating and compressing the log depending on a period

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE log4j:configuration SYSTEM "log4j.dtd">
<log4j:configuration debug="false" xmlns:log4j="http://jakarta.apache.org/log4j/">
  <appender name="CONSOLE" class="org.apache.log4j.ConsoleAppender">
    <layout class="org.apache.log4j.PatternLayout">
      <param name="ConversionPattern" value="%d{yyyy-MM-dd HH:mm:ss,SSS Z} [%-5p] %c - %m%n" />
    </layout>
  </appender>
  <appender name="FILE" class="org.apache.log4j.rolling.RollingFileAppender">
    <param name="Append" value="true" />
    <param name="File" value="${LOGFILE}.log" />
    <rollingPolicy class="org.apache.log4j.rolling.TimeBasedRollingPolicy">
      <param name="ActiveFileName" value="${LOGFILE}.log"/>
      <param name="FileNamePattern" value="${LOGFILE}.%d{yyyyMMdd-hh}.log.gz" />
    </rollingPolicy>
    <layout class="org.apache.log4j.PatternLayout">
      <param name="ConversionPattern" value="%d{yyyy-MM-dd HH:mm:ss,SSS Z} [%-5p] %c - %m%n"/>
    </layout>
  </appender>
  <root>
    <priority value="debug"/>
    <appender-ref ref="FILE"/>
  </root>
</log4j:configuration>
```

The entry `<param name="FileNamePattern" value="${LOGFILE}.%d{yyyyMMdd-hh}.log" />` specifies the rotation frequency:

Frequency	Configuration Sample
Rotate every minute	<code><param name="FileNamePattern" value="\${LOGFILE}.%d{yyyyMMdd-hhmm}.log" /></code>
Rotate every hour	<code><param name="FileNamePattern" value="\${LOGFILE}.%d{yyyyMMdd-hh}.log" /></code>
Rotate every day	<code><param name="FileNamePattern" value="\${LOGFILE}.%d{yyyyMMdd}.log" /></code>

Rotating and compressing the log depending on the size

Rotating and compressing the log depending on the size

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE log4j:configuration SYSTEM "log4j.dtd">
<log4j:configuration debug="false" xmlns:log4j="http://jakarta.apache.org/log4j/">
  <appender name="CONSOLE" class="org.apache.log4j.ConsoleAppender">
    <layout class="org.apache.log4j.PatternLayout">
      <param name="ConversionPattern" value="%d{yyyy-MM-dd HH:mm:ss,SSS Z} [%-5p] %c - %m%n" />
    </layout>
  </appender>
  <appender name="FILE" class="org.apache.log4j.rolling.RollingFileAppender">
    <param name="Append" value="true" />
    <param name="File" value="{LOGFILE}.log" />

    <rollingPolicy class="org.apache.log4j.rolling.FixedWindowRollingPolicy">
      <param name="ActiveFileName" value="{LOGFILE}.log"/>
      <param name="maxIndex" value="5"/>
      <param name="FileNamePattern" value="{LOGFILE}.%i.log.gz"/>
    </rollingPolicy>
    <triggeringPolicy class="org.apache.log4j.rolling.SizeBasedTriggeringPolicy">
      <param name="maxFileSize" value="2048"/>
    </triggeringPolicy>

    <layout class="org.apache.log4j.PatternLayout">
      <param name="ConversionPattern" value="%d{yyyy-MM-dd HH:mm:ss,SSS Z} [%-5p] %c - %m%n"/>
    </layout>
  </appender>
  <root>
    <priority value="debug"/>
    <appender-ref ref="FILE"/>
  </root>
</log4j:configuration>
```

This will create a new file `.n.log` ($n=1..5$) whenever the size exceeds 20KB. You also can specify 20MB etc. The max value for n is 13.