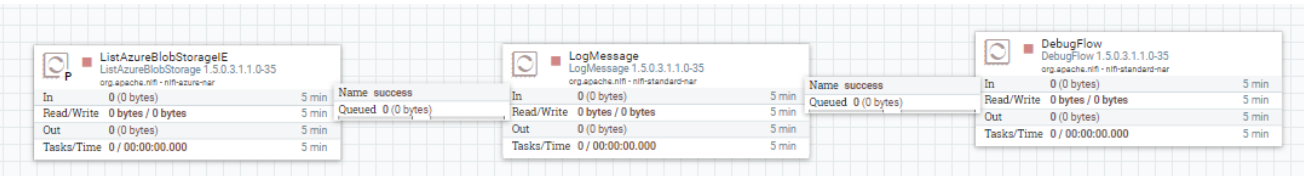


NiFi LogAttribute and LogMessage Processors

LogMessage Processor

We are using following NiFi flow:



Running the `ListAzureBlobStorage` processor, the FlowFiles have following attributes:

```
azure.blobname: AMS_20180619_c7286697-5b62-4eb7-89bb-45b6f7e9b98f.xml
azure.blobtype: Block
azure.etag: "0x8D62535D21D76FC"
azure.length: 1216673278
azure.primaryUri:
https://sdpamdata.blob.core.windows.net/amdata/SDP/UAT/DATA_LAKE/LANDING_ZONE/NLECXUDTG
W01/AMS-LOG/IE/INBOX/AMS_20180619_c7286697-5b62-4eb7-89bb-45b6f7e9b98f.xml
azure.secondaryUri: https://sdpamdata-
secondary.blob.core.windows.net/amdata/SDP/UAT/DATA_LAKE/LANDING_ZONE/NLECXUDTGW01/AMS-
LOG/IE/INBOX/AMS_20180619_c7286697-5b62-4eb7-89bb-45b6f7e9b98f.xml
azure.timestamp: 1538134240000
filename: 1211662732599722
mime.type: application/octet-stream
path: ./
uuid: 8a4b0dba-514b-4743-b974-b845f6db90b1
```

The `LogMessage` processor configuration is as follows:

Property	Value
Log Level	warn
Log prefix	SDP-LOG:
Log message	{"dataflow":"IngestAMSLG","step":"ListAzureBlob","level":"INFO","filename": "\${azure.blobname}","length":\${azure.length},"timestamp":\${azure.timestamp},"flowfile": "\${filename}","uuid":"\${uuid}"}

Once we start the flow, we can check the messages in NiFi log file (`<nifi-root>/logs/nifi-app.log`)

For example, to extract messages, produced by above setup:

```
ssh 10.102.1.6 'grep SDP-LOG: /usr/hdf/current/nifi/logs/nifi-app.log'
```

Example log message looks like this:

```
2018-10-10 16:01:55,075 WARN [Timer-Driven Process Thread-5]
o.a.nifi.processors.standard.LogMessage LogMessage[id=5e91fb3a-0166-1000-0000-
00003ffaa7c8] SDP-LOG:
{"dataflow":"IngestAMSLOG","step":"ListAzureBlob","level":"INFO","filename":
"AMS_20180915_b50effcb-992a-4d83-8cde-
4bc5ab3d1a76.xml","length":1890067336,"timestamp":1538134301000,"flowfile":"12139676870
94216","uuid":"9f0a393a-17c9-4c32-a562-6810584724b7"}
```

This can be converted to a multi-line JSON, using `sed` :

```
ssh 10.102.1.6 'grep SDP-LOG: /usr/hdf/current/nifi/logs/nifi-app.log' | sed 's/^\.*SDP-
LOG:/'
```

The result is:

```
{"dataflow":"IngestAMSLOG","step":"ListAzureBlob","level":"INFO","filename":
"AMS_20180913_ae10812c-a9e2-4c0d-ac30-
eda41dd43dc5.xml","length":1128548316,"timestamp":1538134300000,"flowfile":"12139676869
38618","uuid":"79c17056-2e41-4753-9c03-a29ed4b0d98e"}
{"dataflow":"IngestAMSLOG","step":"ListAzureBlob","level":"INFO","filename":
"AMS_20180914_0270a615-a242-417c-af1f-
66537ca0e332.xml","length":511744506,"timestamp":1538134301000,"flowfile":"121396768698
9917","uuid":"69cc4a3b-d75e-4ccf-b772-6104ada490ae"}
{"dataflow":"IngestAMSLOG","step":"ListAzureBlob","level":"INFO","filename":
"AMS_20180913_ef9a0b0d-b63c-404e-b1b0-
2104fc77d1df.xml","length":1631001725,"timestamp":1538134300000,"flowfile":"12139676869
78817","uuid":"ee76612c-4e08-4e0e-a648-afaeba0745fc"}
{"dataflow":"IngestAMSLOG","step":"ListAzureBlob","level":"INFO","filename":
"AMS_20180914_6da1f7e8-4196-4c81-b64d-
073af8c627d7.xml","length":1612215925,"timestamp":1538134301000,"flowfile":"12139676870
13017","uuid":"f4d338e3-5d0e-4f9b-8a04-ffc289924ecb"}
```

LogAttribute Processor

Using similar setup, but replacing `LogMessage` with `LogAttribute` processor, we can observe following result:

```
2018-10-10 14:08:30,475 INFO [Timer-Driven Process Thread-3]
o.a.n.processors.standard.LogAttribute LogAttribute[id=5e4b6395-0166-1000-ffff-
ffffe89f8994] loggi
ng for flow file StandardFlowFileRecord[uuid=cde52417-b973-44c2-9b29-
e3b8d7cf8d6f,claim=,offset=0,name=1207169337826096,size=0]
-----ListAzureBlobStorageIE: -----
Standard FlowFile Attributes
Key: 'entryDate'
    value: 'Wed Oct 10 14:08:29 UTC 2018'
Key: 'lineageStartDate'
    value: 'Wed Oct 10 14:08:29 UTC 2018'
```

```

Key: 'filesize'
    value: '0'
FlowFile Attribute Map Content
Key: 'azure.blobname'
    value: 'AMS_20180618_1b98541e-5447-4804-9c41-5b358817fb92.xml'
Key: 'azure.blobtype'
    value: 'Block'
Key: 'azure.etag'
    value: '"0x8D62535D1C184C0"'
Key: 'azure.length'
    value: '90676882'
Key: 'azure.primaryUri'
    value:
'https://sdpamdata.blob.core.windows.net/amdata/SDP/UAT/DATA_LAKE/LANDING_ZONE/NLECXUDT
GW01/AMS-LOG/IE/INBOX/AMS_20180618_1b98541e-5447-4804-9c4
1-5b358817fb92.xml'
Key: 'azure.secondaryUri'
    value: 'https://sdpamdata-
secondary.blob.core.windows.net/amdata/SDP/UAT/DATA_LAKE/LANDING_ZONE/NLECXUDTGW01/AMS-
LOG/IE/INBOX/AMS_20180618_1b98541e-544
7-4804-9c41-5b358817fb92.xml'
Key: 'azure.timestamp'
    value: '1538134240000'
Key: 'filename'
    value: '1207169337826096'
Key: 'mime.type'
    value: 'application/octet-stream'
Key: 'path'
    value: './'
Key: 'uuid'
    value: 'cde52417-b973-44c2-9b29-e3b8d7cf8d6f'
-----ListAzureBlobStorageIE: -----

```

Processor settings are as follows:

Property	Value
Log Level	info
Log Payload	false
Attributes to Log	
Attributes to Log by Regular Expression	.*
Attributes to Ignore	
Attributes to Ignore by Regular Expression	
Log prefix	ListAzureBlobStorageIE:
Character Set	UTF-8

Log Configuration

The default NiFi settings

General Notes

`WARN` and `ERROR` Messages are also available in the Bulletin board.

Some Fun

```
ssh 10.102.1.6
grep ListAzureBlobStorageIE /usr/hdf/current/nifi/logs/nifi-app.log

ssh 10.102.1.6 'grep ListAzureBlobStorageIE /usr/hdf/current/nifi/logs/nifi-app.log'

ssh 10.102.1.6 'grep ListAzureBlobStorageIE /usr/hdf/current/nifi/logs/nifi-app.log'
nifi-app_2018-10-10_14.0.log.gz
cat nifi-app_2018-10-10_14.0.log.gz | gzip -d | grep ListAzureBlobStorageIE
```

How to Find the Message in Log File?

NiFi very often does log rotation. Here is how to browse archived log file, using `less`

```
cat nifi-app_2018-10-10_14.0.log.gz | gzip -d | less
```

When less is loaded:

```
/ListAzureBlobStorageIE
```

Resources

- <https://community.hortonworks.com/articles/106931/nifi-debugging-tutorial.html>
- <https://community.hortonworks.com/questions/63071/in-apache-ni-fi-how-can-i-log-all-the-flowfile-att.html>

Appendix: Example LogMessage Output

```
2018-10-10 14:20:14,843 INFO [Timer-Driven Process Thread-8]
o.a.nifi.processors.standard.LogMessage LogMessage[id=3f55432d-db4d-1f19-0000-
0000236127d3] Start:I am here
```

Appendix: Example LogAttribute Output

```
2018-10-10 14:08:30,475 INFO [Timer-Driven Process Thread-3]
o.a.n.processors.standard.LogAttribute LogAttribute[id=5e4b6395-0166-1000-ffff-
ffffe89f8994] loggi
ng for flow file StandardFlowFileRecord[uuid=cde52417-b973-44c2-9b29-
e3b8d7cf8d6f,claim=,offset=0,name=1207169337826096,size=0]
-----ListAzureBlobStorageIE: -----
Standard FlowFile Attributes
Key: 'entryDate'
    Value: 'Wed Oct 10 14:08:29 UTC 2018'
Key: 'lineageStartDate'
    Value: 'Wed Oct 10 14:08:29 UTC 2018'
Key: 'filesize'
    Value: '0'
FlowFile Attribute Map Content
Key: 'azure.blobname'
    Value: 'AMS_20180618_1b98541e-5447-4804-9c41-5b358817fb92.xml'
Key: 'azure.blobtype'
    Value: 'Block'
Key: 'azure.etag'
    Value: '"0x8D62535D1C184C0"'
Key: 'azure.length'
    Value: '90676882'
Key: 'azure.primaryUri'
    Value:
'https://sdpamdata.blob.core.windows.net/amdata/SDP/UAT/DATA_LAKE/LANDING_ZONE/NLECXUDT
GW01/AMS-LOG/IE/INBOX/AMS_20180618_1b98541e-5447-4804-9c4
1-5b358817fb92.xml'
Key: 'azure.secondaryUri'
    Value: 'https://sdpamdata-
secondary.blob.core.windows.net/amdata/SDP/UAT/DATA_LAKE/LANDING_ZONE/NLECXUDTGW01/AMS-
LOG/IE/INBOX/AMS_20180618_1b98541e-544
7-4804-9c41-5b358817fb92.xml'
Key: 'azure.timestamp'
    Value: '1538134240000'
Key: 'filename'
    Value: '1207169337826096'
Key: 'mime.type'
    Value: 'application/octet-stream'
Key: 'path'
    Value: './'
Key: 'uuid'
    Value: 'cde52417-b973-44c2-9b29-e3b8d7cf8d6f'
-----ListAzureBlobStorageIE: -----
```

Appendix. Logging configuration

Here is the observed content `logback.xml` content.

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
  Licensed to the Apache Software Foundation (ASF) under one or more
  contributor license agreements.  See the NOTICE file distributed with
  this work for additional information regarding copyright ownership.
  The ASF licenses this file to You under the Apache License, Version 2.0
  (the "License"); you may not use this file except in compliance with
  the License.  You may obtain a copy of the License at
    http://www.apache.org/licenses/LICENSE-2.0
  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.
-->

<configuration scan="true" scanPeriod="30 seconds">
  <contextListener class="ch.qos.logback.classic.jul.LevelChangePropagator">
    <resetJUL>true</resetJUL>
  </contextListener>

  <appender name="APP_FILE" class="ch.qos.logback.core.rolling.RollingFileAppender">
    <file>${org.apache.nifi.bootstrap.config.log.dir}/nifi-app.log</file>
    <rollingPolicy
class="ch.qos.logback.core.rolling.SizeAndTimeBasedRollingPolicy">
      <!--
        For daily rollover, use 'app_%d.log'.
        For hourly rollover, use 'app_%d{yyyy-MM-dd_HH}.log'.
        To GZIP rolled files, replace '.log' with '.log.gz'.
        To ZIP rolled files, replace '.log' with '.log.zip'.
      -->
      <fileNamePattern>${org.apache.nifi.bootstrap.config.log.dir}/nifi-
app_%d{yyyy-MM-dd_HH}_%i.log.gz</fileNamePattern>
      <maxFileSize>100MB</maxFileSize>
      <!-- keep 30 log files worth of history -->
      <maxHistory>30</maxHistory>
    </rollingPolicy>
    <immediateFlush>true</immediateFlush>
    <encoder class="ch.qos.logback.classic.encoder.PatternLayoutEncoder">
      <pattern>%date %level [%thread] %logger{40} %msg%n</pattern>
    </encoder>
  </appender>

  <appender name="USER_FILE" class="ch.qos.logback.core.rolling.RollingFileAppender">
    <file>${org.apache.nifi.bootstrap.config.log.dir}/nifi-user.log</file>
    <rollingPolicy class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">
```

```

    <!--
        For daily rollover, use 'user_%d.log'.
        For hourly rollover, use 'user_%d{yyyy-MM-dd_HH}.log'.
        To GZIP rolled files, replace '.log' with '.log.gz'.
        To ZIP rolled files, replace '.log' with '.log.zip'.
    -->
    <fileNamePattern>${org.apache.nifi.bootstrap.config.log.dir}/nifi-
user_%d.log.gz</fileNamePattern>
    <!-- keep 30 log files worth of history -->
    <maxHistory>30</maxHistory>
</rollingPolicy>
<encoder class="ch.qos.logback.classic.encoder.PatternLayoutEncoder">
    <pattern>%date %level [%thread] %logger{40} %msg%n</pattern>
</encoder>
</appender>

<appender name="BOOTSTRAP_FILE"
class="ch.qos.logback.core.rolling.RollingFileAppender">
    <file>${org.apache.nifi.bootstrap.config.log.dir}/nifi-bootstrap.log</file>
    <rollingPolicy class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">
    <!--
        For daily rollover, use 'user_%d.log'.
        For hourly rollover, use 'user_%d{yyyy-MM-dd_HH}.log'.
        To GZIP rolled files, replace '.log' with '.log.gz'.
        To ZIP rolled files, replace '.log' with '.log.zip'.
    -->
    <fileNamePattern>${org.apache.nifi.bootstrap.config.log.dir}/nifi-
bootstrap_%d.log.gz</fileNamePattern>
    <!-- keep 5 log files worth of history -->
    <maxHistory>5</maxHistory>
</rollingPolicy>
<encoder class="ch.qos.logback.classic.encoder.PatternLayoutEncoder">
    <pattern>%date %level [%thread] %logger{40} %msg%n</pattern>
</encoder>
</appender>

<appender name="CONSOLE" class="ch.qos.logback.core.ConsoleAppender">
    <encoder class="ch.qos.logback.classic.encoder.PatternLayoutEncoder">
        <pattern>%date %level [%thread] %logger{40} %msg%n</pattern>
    </encoder>
</appender>

<!-- valid logging levels: TRACE, DEBUG, INFO, WARN, ERROR -->

<logger name="org.apache.nifi" level="INFO"/>
<logger name="org.apache.nifi.processors" level="WARN"/>
<logger name="org.apache.nifi.processors.standard.LogAttribute" level="INFO"/>
<logger name="org.apache.nifi.processors.standard.LogMessage" level="INFO"/>
<logger name="org.apache.nifi.controller.repository.StandardProcessSession"
level="WARN" />

<logger name="org.apache.zookeeper.ClientCnxn" level="ERROR" />

```

```

<logger name="org.apache.zookeeper.server.NIOServerCnxn" level="ERROR" />
<logger name="org.apache.zookeeper.server.NIOServerCnxnFactory" level="ERROR" />
<logger name="org.apache.zookeeper.server.quorum" level="ERROR" />
<logger name="org.apache.zookeeper.ZooKeeper" level="ERROR" />
<logger name="org.apache.zookeeper.server.PreRequestProcessor" level="ERROR" />

<logger name="org.apache.calcite.runtime.CalciteException" level="OFF" />

<logger name="org.apache.curator.framework.recipes.leader.LeaderSelector"
level="OFF" />
<logger name="org.apache.curator.ConnectionState" level="OFF" />

<!-- Logger for managing logging statements for nifi clusters. -->
<logger name="org.apache.nifi.cluster" level="INFO"/>

<!-- Logger for logging HTTP requests received by the web server. -->
<logger name="org.apache.nifi.server.JettyServer" level="INFO"/>

<!-- Logger for managing logging statements for jetty -->
<logger name="org.eclipse.jetty" level="INFO"/>

<!-- Suppress non-error messages due to excessive logging by class or library -->
<logger name="org.springframework" level="ERROR"/>

<!-- Suppress non-error messages due to known warning about redundant path
annotation (NIFI-574) -->
<logger name="org.glassfish.jersey.internal.Errors" level="ERROR"/>

<!--
    Logger for capturing user events. We do not want to propagate these
    log events to the root logger. These messages are only sent to the
    user-log appender.
-->
<logger name="org.apache.nifi.web.security" level="INFO" additivity="false">
    <appender-ref ref="USER_FILE"/>
</logger>
<logger name="org.apache.nifi.web.api.config" level="INFO" additivity="false">
    <appender-ref ref="USER_FILE"/>
</logger>
<logger name="org.apache.nifi.authorization" level="INFO" additivity="false">
    <appender-ref ref="USER_FILE"/>
</logger>
<logger name="org.apache.nifi.cluster.authorization" level="INFO"
additivity="false">
    <appender-ref ref="USER_FILE"/>
</logger>
<logger name="org.apache.nifi.web.filter.RequestLogger" level="INFO"
additivity="false">
    <appender-ref ref="USER_FILE"/>
</logger>

<!--

```



```
    Logger for capturing Bootstrap logs and NiFi's standard error and standard out.
-->
<logger name="org.apache.nifi.bootstrap" level="INFO" additivity="false">
    <appender-ref ref="BOOTSTRAP_FILE" />
</logger>
<logger name="org.apache.nifi.bootstrap.Command" level="INFO" additivity="false">
    <appender-ref ref="CONSOLE" />
    <appender-ref ref="BOOTSTRAP_FILE" />
</logger>

<!-- Everything written to NiFi's Standard Out will be logged with the logger
org.apache.nifi.Stdout at INFO level -->
<logger name="org.apache.nifi.Stdout" level="INFO" additivity="false">
    <appender-ref ref="BOOTSTRAP_FILE" />
</logger>

<!-- Everything written to NiFi's Standard Error will be logged with the logger
org.apache.nifi.Stderr at ERROR level -->
<logger name="org.apache.nifi.Stderr" level="ERROR" additivity="false">
    <appender-ref ref="BOOTSTRAP_FILE" />
</logger>

<root level="INFO">
    <appender-ref ref="APP_FILE"/>
</root>

</configuration>
```