

Flume mit LZO

Voraussetzung für diese Anleitung ist, dass ihr Flume und Hadoop LZO bereits wie geschildert installiert habt:

- [\[wiki\]Hadoop mit LZO\[/wiki\]](#)
- [\[wiki\]Cloudera Flume Installation\[/wiki\]](#)

== LZO Bibliotheken ==

Wir müssen die LZO Dateien von Hadoop auch unter Flume bekannt machen. Beachtet, dass ihr im folgenden vielleicht den Ordernamen Linux-amd64-64 ersetzen müsst.

Quellcode

1. # Copy the jar file
2. `sudo cp /usr/lib/hadoop-0.20/lib/hadoop-lzo-*.jar /usr/lib/flume/lib/`
3. # Copy the native library
5. `sudo mkdir -p /usr/lib/flume/lib/native/Linux-amd64-64`
6. `sudo cp /usr/lib/hadoop-0.20/lib/native/Linux-amd64-64/*compression* /usr/lib/flume/lib/native/Linux-amd64-64`

== LZO Bibliotheken bekannt geben ==

Damit Flume die nativen LZO Erweiterungen lädt, muss der native Ordner noch über den Java Library Pfad bekannt gegeben werden.

Dazu legen wir die Datei `/usr/lib/flume/bin/flume-env.sh` an, falls sie noch nicht existiert und fügen folgenden Inhalt hinzu.

Auch hier muss wieder darauf geachtet werden, dass die richtige Architektur verwendet wird.

Quellcode

1. `export JAVA_LIBRARY_PATH=/usr/lib/flume/lib/native/Linux-amd64-64`

== LZO Konfiguration ==

Nun muss LZO noch in der Flume Konfiguration aktiviert werden. Falls unter `/etc/flume/conf/flume-site.xml` noch keine Konfiguration vorhanden ist, dann orientiert euch an der Beispielkonfiguration unter `/etc/flume/conf/flume-site.xml.template`.

Folgende Einstellung muss hinzugefügt werden:

Quellcode

1. `<property>`
2. `<name>flume.collector.dfs.compress.codec</name>`
3. `<value>LzopCodec</value>`
4. `<description>Writes formatted data compressed in specified codec to`
5. `dfs. Value is None, GzipCodec, DefaultCodec (deflate), BZip2Codec,`
6. `or any other Codec hadoop is aware of </description>`
7. `</property>`

Danach solltet ihr Flume neustarten:

Quellcode

1. `for service in /etc/init.d/flume-*; do sudo $service restart; done`

== [Flume mit PHP](#) ==

Solltet ihr auch der Anleitung [\[wiki\]Flume mit PHP\[/wiki\]](#) gefolgt sein, dann könnt ihr die LZO komprimierten Sequence

Files auch über LZO einsehen.

Sie sollten dann wie im Screenshot aussehen:

[easy-coding.de/Attachment/1123...b39f9d6c566b6707046908ad5](https://www.easy-coding.de/Attachment/1123...b39f9d6c566b6707046908ad5)

== Alte Flume Version? ==

Die Flume Version <= 0.9.3-CDH3B4 enthält noch Fehler im Umgang mit der Kompression.

In der [Flume User Group](#) wurden die Probleme erörtert und gelöst.

Ich beschreibe den Workaround hier, der beschreibt wie man die entsprechenden Dateien aus den Git Quellen patcht.

Quellcode

1. `git clone https://github.com/cloudera/flume.git`
2. `cd flume`
3. `ant`
4. `sudo cp bin/flume /usr/lib/flume/bin/flume`
5. `sudo cp build/flume-*-test.jar /usr/lib/flume/lib/flume-*-test.jar`
6. `sudo cp build/flume-*-core.jar /usr/lib/flume/lib/flume-*-core.jar`