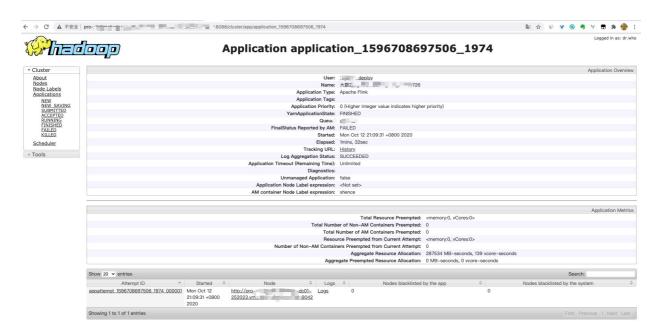
前言





对于这种我们页面我们只能查看 JobManager 的日志,不再可以查看作业挂掉之前的运行的 Web UI,很难清楚知道作业在挂的那一刻到底发生了啥?如果我们还没有 Metrics 监控的话,那么完全就只能通过日志去分析和定位问题了,所以如果能还原之前的 Web UI,我们可以通过 UI 发现和定位一些问题。

History Server

那么这里就需要利用 Flink 中的 History Server 来解决这个问题,如何开启这个呢?

你需要在 flink-conf.yml 中配置如下:

```
=====
    # HistoryServer
 4
    # The HistoryServer is started and stopped via bin/historyserver.sh
    (start|stop)
 6
    # Directory to upload completed jobs to. Add this directory to the list of
   # monitored directories of the HistoryServer as well (see below).
8
   # flink job 运行完成后的日志存放目录
10
   jobmanager.archive.fs.dir: hdfs:///flink/history-log
11
12
   # The address under which the web-based HistoryServer listens.
   # flink history进程所在的主机
```

```
14
   #historyserver.web.address: 0.0.0.0
15
16
   # The port under which the web-based HistoryServer listens.
   # flink history进程的占用端口
17
   #historyserver.web.port: 8082
18
19
20
   # Comma separated list of directories to monitor for completed jobs.
21
   # flink history进程的hdfs监控目录
22 historyserver.archive.fs.dir: hdfs:///flink/history-log
2.3
   # Interval in milliseconds for refreshing the monitored directories.
24
25 # 刷新受监视目录的时间间隔(以毫秒为单位)
26 #historyserver.archive.fs.refresh-interval: 10000
```

注意: jobmanager.archive.fs.dir 要和 historyserver.archive.fs.dir 配置的路径要一样

执行命令:

1 ./bin/historyserver.sh start

发现报错如下:

```
1 2020-10-13 21:21:01,310 main INFO org.apache.flink.core.fs.FileSystem
                          - Hadoop is not in the classpath/dependencies. The
    extended set of supported File Systems via Hadoop is not available.
   2020-10-13 21:21:01,336 main INFO
    org.apache.flink.runtime.security.modules.HadoopModuleFactory - Cannot
    create Hadoop Security Module because Hadoop cannot be found in the
    Classpath.
   2020-10-13 21:21:01,352 main INFO
    org.apache.flink.runtime.security.modules.JaasModule
                                                                  - Jaas file
    will be created as /tmp/jaas-354359771751866787.conf.
   2020-10-13 21:21:01,355 main INFO
    org.apache.flink.runtime.security.SecurityUtils
    install HadoopSecurityContext because Hadoop cannot be found in the
    Classpath.
   2020-10-13 21:21:01,363 main WARN
    org.apache.flink.runtime.webmonitor.history.HistoryServer
                                                                   - Failed to
    create Path or FileSystem for directory 'hdfs:///flink/history-log'.
    Directory will not be monitored.
   org.apache.flink.core.fs.UnsupportedFileSystemSchemeException: Could not
    find a file system implementation for scheme 'hdfs'. The scheme is not
    directly supported by Flink and no Hadoop file system to support this
    scheme could be loaded.
            at.
    org.apache.flink.core.fs.FileSystem.getUnguardedFileSystem(FileSystem.java
    :450)
            at org.apache.flink.core.fs.FileSystem.get(FileSystem.java:362)
9
            at org.apache.flink.core.fs.Path.getFileSystem(Path.java:298)
           at org.apache.flink.runtime.webmonitor.history.HistoryServer.
    <init>(HistoryServer.java:187)
11
            at org.apache.flink.runtime.webmonitor.history.HistoryServer.
    <init>(HistoryServer.java:137)
12
    org.apache.flink.runtime.webmonitor.history.HistoryServer$1.call(HistorySe
    rver.java:122)
```

```
13
    org.apache.flink.runtime.webmonitor.history.HistoryServer$1.call(HistorySe
    rver.java:119)
14
            at
    org.apache.flink.runtime.security.NoOpSecurityContext.runSecured(NoOpSecur
    ityContext.java:30)
15
            at
    org.apache.flink.runtime.webmonitor.history.HistoryServer.main(HistoryServ
    Caused by: org.apache.flink.core.fs.UnsupportedFileSystemSchemeException:
    Hadoop is not in the classpath/dependencies.
17
            at
    org.apache.flink.core.fs.UnsupportedSchemeFactory.create(UnsupportedScheme
    Factory.java:58)
18
    org.apache.flink.core.fs.FileSystem.getUnguardedFileSystem(FileSystem.java
    :446)
            ... 8 more
20
   2020-10-13 21:21:01,367 main ERROR
    org.apache.flink.runtime.webmonitor.history.HistoryServer
                                                                - Failed to
    run HistoryServer.
21
    org.apache.flink.util.FlinkException: Failed to validate any of the
    configured directories to monitor.
            at org.apache.flink.runtime.webmonitor.history.HistoryServer.
    <init>(HistoryServer.java:196)
            at org.apache.flink.runtime.webmonitor.history.HistoryServer.
23
    <init>(HistoryServer.java:137)
24
    org.apache.flink.runtime.webmonitor.history.HistoryServer$1.call(HistorySe
    rver.java:122)
25
            at
    org.apache.flink.runtime.webmonitor.history.HistoryServer$1.call(HistorySe
    rver.java:119)
26
    org.apache.flink.runtime.security.NoOpSecurityContext.runSecured(NoOpSecur
    ityContext.java:30)
27
    org.apache.flink.runtime.webmonitor.history.HistoryServer.main(HistoryServ
    er.java:119)
```

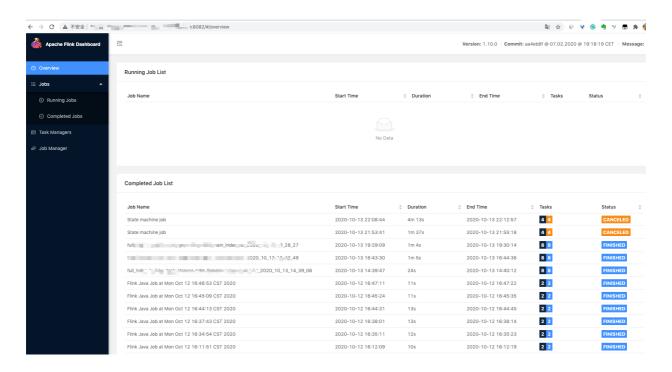
这个异常的原因是因为 Flink 集群的 CLASS_PATH 下缺少了 HDFS 相关的 jar,我们可以引入 HDFS 的依赖放到 lib 目录下面或者添加 Hadoop 的环境变量。

这里我们在 historyserver.sh 脚本中增加下面脚本,目的就是添加 Hadoop 的环境变量:

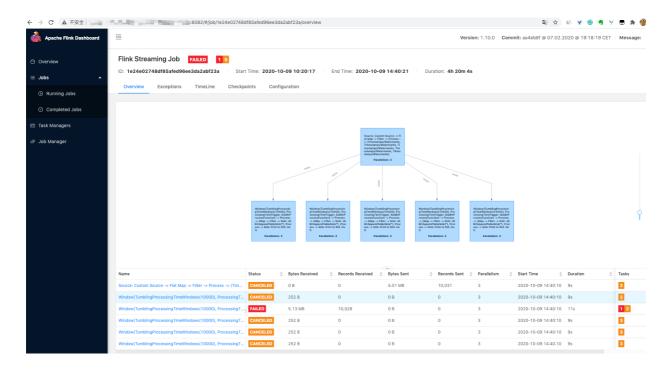
```
# export hadoop classpath
if [ `command -v hadoop` ];then
export HADOOP_CLASSPATH=`hadoop classpath`
else
cho "hadoop command not found in path!"
fi
```

效果

添加后再启动脚本则可以运行成功了,打开页面 机器IP:8082 则可以看到历史所有运行完成或者 失败的作业列表信息。



点进单个作业可以看到作业挂之前的所有信息,便于我们去查看挂之前作业的运行情况(Exception 信息/Checkpoint 信息/算子的流入和流出数据量信息等)



原理分析

再来看看配置的 /flink/history-log/ 目录有什么东西呢? 执行下面命令可以查看

1 hdfs dfs -ls /flink/history-log/

其实 history server 会在本地存储已结束 Job 信息,你可以配置

historyserver.web.tmpdir 来决定存储在哪,默认的拼接规则为:

```
System.getProperty("java.io.tmpdir") + File.separator + "flink-web-history-
" + UUID.randomUUID()
```

Linux 系统临时目录为 /tmp, 你可以看到源码中 HistoryServerOptions 该类中的可选参数。

那么我们找到本地该临时目录,可以观察到里面保存着很多 JS 文件,其实就是我们刚才看到的页面

历史服务存储文件中,存储了用于页面展示的模板配置。历史任务信息存储在 Jobs 路径下,其中包含了已经完成的 Job,每次启动都会从 historyserver.archive.fs.dir 拉取所有的任务元数据信息。

```
-rw-rw-r- 1 deploy deploy drwxrwxr-x 4 deploy deploy deploy deploy drwxrwxr-x 4 deploy deploy deploy drwxrwxr-x 4 deploy deploy deploy deploy drwxrwxr-x 4 deploy d
```

每个任务文件夹中包含我们需要获取的一些信息,通过 REST API 获取时指标时,就是返回这些内容(Checkpoint/Exception 信息等)。

```
$ ll /tmp/flink-web-history-02036d66-be85-4193-907b-1529cad44118/jobs/fff16030ed22c35e2f3436e027cde2a7 total 28
-rw-rw-r-- 1 deploy deploy 90 Oct 13 21:55 accumulators.json
drwxrwxr-x 2 deploy deploy 4096 Oct 13 21:55 checkpoints
-rw-rw-r-- 1 deploy deploy 310 Oct 13 21:55 checkpoints.json
-rw-rw-r-- 1 deploy deploy 351 Oct 13 21:55 config.json
-rw-rw-r-- 1 deploy deploy 78 Oct 13 21:55 exceptions.json
-rw-rw-r-- 1 deploy deploy 315 Oct 13 21:55 plan.json
drwxrwxr-x 3 deploy deploy 4096 Oct 13 21:55 vertices
```

REST API

以下是可用且带有示例 JSON 响应的请求列表。所有请求格式样例均为

http://hostname:8082/jobs,下面我们仅列出了 URLs 的 *path* 部分。 尖括号中的值为变量,例如作业 7684be6004e4e955c2a558a9bc463f65 的

http://hostname:port/jobs/<jobid>/exceptions 请求须写为

http://hostname:port/jobs/7684be6004e4e955c2a558a9bc463f65/exceptions.

- /config
- /jobs/overview
- o /jobs/<jobid>
- o /jobs/<jobid>/vertices
- /jobs/<jobid>/config

- /jobs/<jobid>/exceptions
- /jobs/<jobid>/accumulators
- o /jobs/<jobid>/vertices/<vertexid>
- /jobs/<jobid>/vertices/<vertexid>/subtasktimes
- /jobs/<jobid>/vertices/<vertexid>/taskmanagers
- /jobs/<jobid>/vertices/<vertexid>/accumulators
- /jobs/<jobid>/vertices/<vertexid>/subtasks/accumulators
- /jobs/<jobid>/vertices/<vertexid>/subtasks/<subtasknum>
- /jobs/<jobid>/vertices/<vertexid>/subtasks/<subtasknum>/attempts /<attempt>
- /jobs/<jobid>/vertices/<vertexid>/subtasks/<subtasknum>/attempts /<attempt>/accumulators
- /jobs/<jobid>/plan

总结

这样我们就可以开心的去查看作业挂之前的 Web UI 信息了,妈妈在也不用担心我的作业挂了! 😁



参考文章

- History Server
- 。 flink历史服务