DATA+AI
SUMMIT 2022

Obfuscating Sensitive Information from Spark UI and Logs

Yian Liou

Software Engineer, Workday

ORGANIZED BY databricks

Agenda

- Quick overview
- Obfuscating string representations of column names
- Obfuscating the logging from 3rd party libraries
- Summary

Background Context

- Workday Prism Analytics uses Spark for data processing on customer data for Finance and HR use cases.
- Spark UI/logs are useful for debugging, but we can't show it as-is because of schema information leakage.

Goals

- Obfuscate column names on the UI and in logs [SPARK-37610] to be able to share anonymized spark plans while debugging.
- Add custom log4j appender to control which third party library log messages are and aren’t obfuscated.

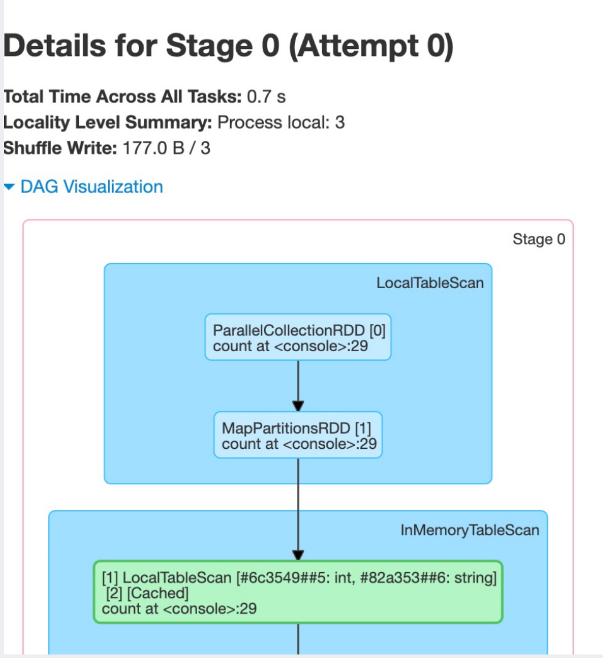
Obfuscating Column Names

- Implemented via adding ObfuscatedString method in classes such as Treenode.Scala to hash column names.
- Obfuscate other fields such as Alias names, expressions, and string constants
- Added a config to use ObfuscatedString method when showing information on the UI and logs.

Obfuscating the logging from 3rd party libraries

- Use a custom log4j appender by adding an ObfuscatingAppender class and a configuration file for whitelisting log lines from external 3rd party jars.
- This uses a list of rules based on class name, package name, log message regexes to decide whether to obfuscate third party libraries or not.

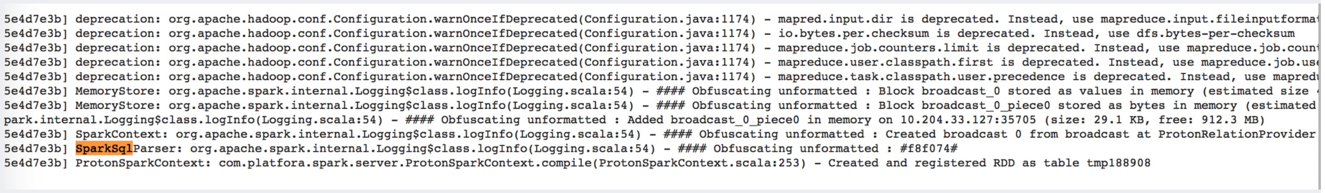
Results



Details for Stage 0 (Attempt 0)

Total Time Across All Tasks: 0.7 s
Locality Level Summary: Process local: 3
Shuffle Write: 177.0 B / 3

▾ DAG Visualization



```
5e4d7e3b] deprecation: org.apache.hadoop.conf.Configuration.warnOnceIfDeprecated(Configuration.java:1174) - mapred.input.dir is deprecated. Instead, use mapreduce.input.fileinputforma
5e4d7e3b] deprecation: org.apache.hadoop.conf.Configuration.warnOnceIfDeprecated(Configuration.java:1174) - io.bytes.per.checksum is deprecated. Instead, use dfs.bytes-per-checksum
5e4d7e3b] deprecation: org.apache.hadoop.conf.Configuration.warnOnceIfDeprecated(Configuration.java:1174) - mapreduce.job.counters.limit is deprecated. Instead, use mapreduce.job.coun
5e4d7e3b] deprecation: org.apache.hadoop.conf.Configuration.warnOnceIfDeprecated(Configuration.java:1174) - mapreduce.user.classpath.first is deprecated. Instead, use mapreduce.job.us
5e4d7e3b] deprecation: org.apache.hadoop.conf.Configuration.warnOnceIfDeprecated(Configuration.java:1174) - mapreduce.task.classpath.user.precedence is deprecated. Instead, use mapred
5e4d7e3b] MemoryStore: org.apache.spark.internal.Logging$class.logInfo(Logging.scala:54) - #### Obfuscating unformatted : Block broadcast_0 stored as values in memory (estimated size
5e4d7e3b] MemoryStore: org.apache.spark.internal.Logging$class.logInfo(Logging.scala:54) - #### Obfuscating unformatted : Block broadcast_0_piece0 stored as bytes in memory (estimated
park.internal.Logging$class.logInfo(Logging.scala:54) - #### Obfuscating unformatted : Added broadcast_0_piece0 in memory on 10.204.33.127:35705 (size: 29.1 KB, free: 912.3 MB)
5e4d7e3b] SparkContext: org.apache.spark.internal.Logging$class.logInfo(Logging.scala:54) - #### Obfuscating unformatted : Created broadcast 0 from broadcast at ProtonRelationProvider
5e4d7e3b] SparkSqlParser: org.apache.spark.internal.Logging$class.logInfo(Logging.scala:54) - #### Obfuscating unformatted : #f8f074#
5e4d7e3b] ProtonSparkContext: com.platfora.spark.server.ProtonSparkContext.compile(ProtonSparkContext.scala:253) - Created and registered RDD as table tmp188908
```

```
[1] Project [coalesce(CASE WHEN #39ae27# THEN #8a7988# ELSE #939029# END, #0ef821#) AS #d993de##11320]
+- [2] SubqueryAlias `src`
   +- [3] Project [#46c9e2##11313: int AS #2c70e1##11316, #90b794##11314: string AS #cd4240##11317]
      +- [4] LocalRelation [#46c9e2##11313: int, #90b794##11314: string]
```

Summary

- Obfuscating column names along with adding custom log4j appender helped us to protect customer’s data from leaking in the spark logs and on the UI.
- Alleviates some privacy concerns when sharing Spark plans which can occur when debugging production use cases.

DATA+AI
SUMMIT 2022

Thank you