Hortonworks Cybersecurity Platform

Upgrade Guide

(May 18, 2018)

docs.cloudera.com

Hortonworks Cybersecurity Platform: Upgrade Guide

Hortonworks Cybersecurity Platform (HCP) is a modern data application based on Apache Metron, powered by Apache Hadoop, Apache Storm, and related technologies.

HCP provides a framework and tools to enable greater efficiency in Security Operation Centers (SOCs) along with better and faster threat detection in real-time at massive scale. It provides ingestion, parsing and normalization of fully enriched, contextualized data, threat intelligence feeds, triage and machine learning based detection. It also provides end user near real-time dashboarding.

Based on a strong foundation in the Hortonworks Data Platform (HDP) and Hortonworks DataFlow (HDF) stacks, HCP provides an integrated advanced platform for security analytics.

Please visit the Hortonworks Data Platform page for more information on Hortonworks technology. For more information on Hortonworks services, please visit either the Support or Training page. Feel free to Contact Us directly to discuss your specific needs.

Table of Contents

List of Figures

1. Preparing to Upgrade

Prior to upgrading Hortonworks Cybersecurity Platform (HCP), you must back up your configuration and stop all Metron services.

1.1. Backing up Your Configuration

The HCP upgrade uses the default configuration for the new Metron version. If you made any changes to the Metron configuration in the previous version, you need to back up your old configuration so you can incorporate those changes into the new Metron configuration. You will also need to re-enter values for the Metron properties in Ambari.

1. Create a backup directory.

   ```
   mkdir /$HCP_BACKUP_DIRECTORY
   ```

2. Back up your configuration information in ZooKeeper to your backup directory:

   ```
   ${METRON_HOME}/bin/zk_load_configs.sh -m DUMP -z $ZOOKEEPER > /
   $HCP_BACKUP_DIRECTORY/$BACKUP_CONFIG.txt
   ```

3. Back up the following property files in the `$METRON_HOME/config` directory to your backup directory:

   - elasticsearch.properties
   - enrichment.properties
   - pcap.properties
   - profiler properties

   For example:

   ```
   cp elasticsearch.properties /$HCP_BACKUP/elasticsearch.properties
   ```

4. Copy the zookeeper directory to your backup directory:

   ```
   cp -R zookeeper/ /$HCP_BACKUP/zookeeper
   ```

5. Back up your Metron configuration.

   The easiest way to do this is to take a screenshot of each of the Metron configuration pages that you modified in Ambari. At a minimum, take a screen shot of the following configuration pages:

   - Index Settings
   - Parsers
   - REST

1.2. Stopping All Metron Services

You need to stop all Metron services prior to uninstalling Metron.

1. Stop all Metron services in Ambari.

   Stop each Metron service in the following order:

   - Metron Alerts UI
   - Metron Management UI
   - Metron REST
   - Storm

     To stop Storm, complete the following steps:

     a. Kill each Storm topology.

        From the Storm node, list all of the Storm topologies that are currently running:

        ```
        storm list
        ```

     b. Kill each of the running Storm topologies in the following order:

        - all parsers such as bro and snort
        - enrichment
        - indexing
        - profiler

        For example:

        ```
        storm kill bro
        ```

     c. Return to the Storm UI.

        Verify that all topologies are killed.

     d. In Ambari, stop Storm by selecting Storm and clicking **Stop All** in the **Actions** menu.

2. Ensure that the UIs are shut down.

   If the Metron Alerts Ui or Metron Management UI status in Ambari is "running," shut down the UIs by entering the following from `$METRON_HOME/var/log/metron/metron`:

   ```
   service metron-alerts-ui status
   service metron-alerts-ui stop

   service metron-management-ui status
   service metron-management-ui stop
   ```

2. Upgrading Metron

After you shut down Metron and all of its services, you must uninstall Metron and then reinstall the newest version of Metron.

Prerequisite

- Back up your Metron configuration.

  See Backing up Your Configuration for more information.

- Stop all Metron services

  See Stopping All Metron Services for more information.

1. Uninstall Metron.

   In Ambari, select **Metron**, then under the **Service Actions** menu, click **Delete Service**.

   When prompted, enter "delete" to confirm deleting the service.

2. Remove all of the rpms from the old Metron version.

   CentOS

   a. From the Ambari node, enter the following to list the Metron package name:

      ```
      rpm -qa | grep metron
      ```

      You should see output similar to the following:

      ```
      metron_1_4_2_0_23-config-0.4.1.1.4.2.0-23.noarch
      ```

   b. Enter the following to list all of the Metron packages:

      ```
      sudo rpm -q --scripts  metron_1_4_2_0_23-config-0.4.1.1.4.2.0-23.noarch
      ```

      You should see output similar to the following:

      ```
      chkconfig --add metron-management-ui
      chkconfig --add metron-alerts-ui
      preuninstall scriptlet (using /bin/sh):
      chkconfig --del metron-management-ui
      chkconfig --del metron-alerts-ui
      ```

   c. Remove each of the packages:

      ```
      rpm remove $PACKAGE_NAME
      ```

      For example:

      ```
      sudo chkconfig --del metron-management-ui
      ```

   Ubuntu

   From the Ambari node, enter the following to delete all of the Metron packages:

```
sudo aptitude purge $PACKAGE_NAME
```

3. Modify the `/etc/yum.repos.d/HCP.repo` file with the updated repo version:

   ```
   vi /etc/yum.repos.d/HCP.repo
   ```

4. Update the `HCP.repo` file.

   CentOS

   ```
   yum update
   ```

   Ubuntu

   ```
   apt-get update
   ```

5. Install the current HCP mpack repo from Release Notes.

   For example:

   ```
   wget http://public-repo-1.hortonworks.com/HCP/centos7/1.x/updates/1.5.0.0/
   tars/metron/hcp-ambari-mpack-1.5.0.0-9.tar.gz
   ambari-server install-mpack --force --mpack=/${MPACK_DOWNLOAD_DIRECTORY}/
   hcp-ambari-mpack-1.5.0.0-9.tar.gz --verbose
   ```

6. Restart the Ambari server.

   ```
   ambari-server restart
   ```

7. Re-open Ambari and add back the updated Metron version.

   From the **Actions** menu, click **Add Service**, then click **`Metron`** from the **Choose Services** page. Ensure Metron is the updated version.

   Ambari lists each service on which Metron is dependent.

8. Click **`yes`** to add each dependency.

9. In Ambari, add back your Metron configuration information in the **Property** fields.

   Important

   Do not copy and paste into the Metron property fields. You can inadvertently add a special character.

10.Click **Deploy** to start the Metron set up.

   The process to install, start, and test Metron will take a while.

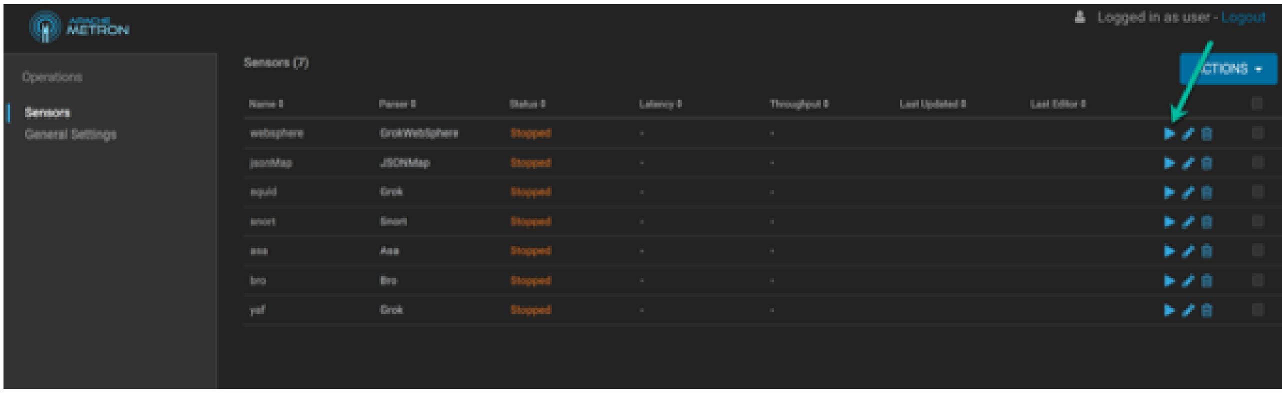
11.Restart the Metron services:

   - Metron REST
   - Metron Management UI
   - Metron Alerts UI

- Indexing

12. In the Management UI, restart the Metron Parsers including Enrichment, Bro, Snort, Yaf, and any other parsers you added previously.

Figure 2.1. Management UI



Sensors (7)

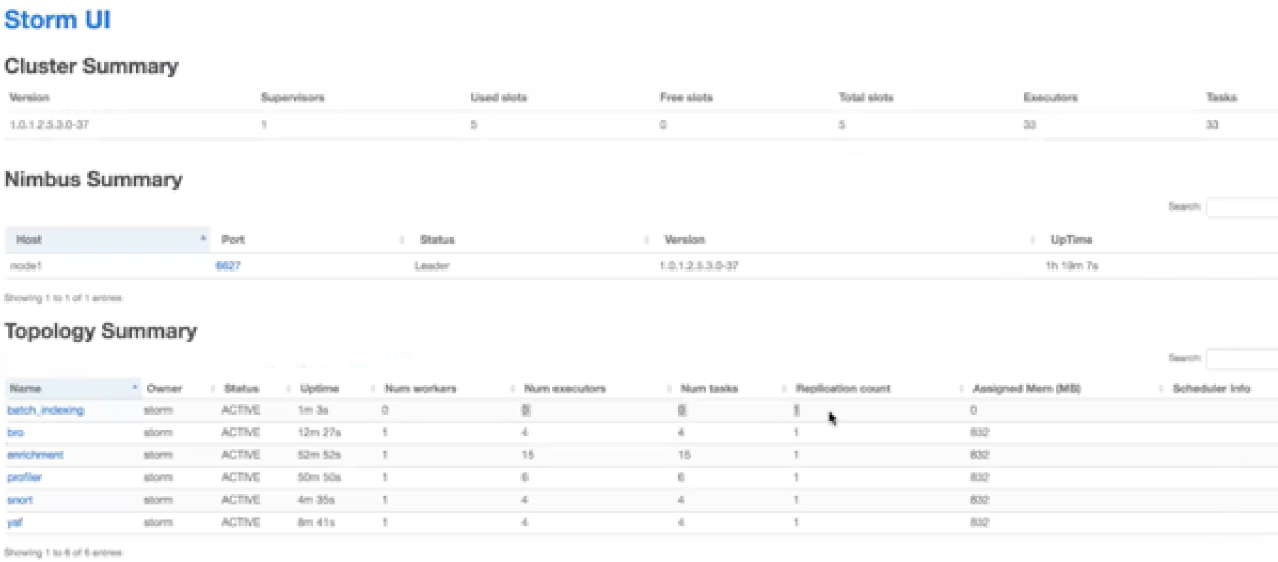
Name	Parser	Status	Latency	Throughput	Last Updated	Last Editor
websphere	GrokWebSphere	Stopped	-	-		
jsonMap	JSONMap	Stopped	-	-		
squid	Grok	Stopped	-	-		
snort	Snort	Stopped	-	-		
asa	Asa	Stopped	-	-		
bro	Bro	Stopped	-	-		
yaf	Grok	Stopped	-	-		

Note

Starting the Metron parsers might take a while.

13. Check the status of the parsers in the Storm UI.

Figure 2.2. Storm UI



Storm UI

Cluster Summary

Version	Supervisors	Used slots	Free slots	Total slots	Executors	Tasks
1.0.1.2.5.3.0-37	1	5	0	5	33	33

Nimbus Summary

Host	Port	Status	Version	UpTime
node1	6627	Leader	1.0.1.2.5.3.0-37	1h 19m 7s

Showing 1 to 1 of 1 entries

Topology Summary

Name	Owner	Status	Uptime	Num workers	Num executors	Num tasks	Replication count	Assigned Mem (MB)	Scheduler Info
batch_indexing	storm	ACTIVE	1m 3s	0	0	0	1	0	
bro	storm	ACTIVE	12m 27s	1	4	4	1	832	
enrichment	storm	ACTIVE	52m 52s	1	15	15	1	832	
profiler	storm	ACTIVE	50m 50s	1	6	6	1	832	
snort	storm	ACTIVE	4m 35s	1	4	4	1	832	
yaf	storm	ACTIVE	8m 41s	1	4	4	1	832	

Showing 1 to 6 of 6 entries

3. Mandatory Post-Upgrade Tasks

After you finish updating the Ambari M-Pack, depending on your configuration, you need to update the following features in your cluster:

- Upgrading Your Configuration [6]
- Changes to STELLAR Language [6]

3.1. Upgrading Your Configuration

The upgrade uses the default configuration for the new Metron version. If you made any changes to the Metron configuration in the previous version, incorporate those changes into the new Metron configuration by changing one or more of the following:

- Metron properties in Ambari
- ZooKeeper

  Incorporate changes from the ZooKeeper file you backed up earlier.
- Flux files

  Incorporate changes from the Flux files you backed up earlier.

3.2. Changes to STELLAR Language

HCP adds additional Stellar keywords to each new HCP version. These new keywords might cause compatablity issues where these reserved words and symbols are used in existing scripts. Be sure to check the Stellar Language Quick Reference for new and changed Stellar keywords.

HCP 1.5.0 adds `match` to the Stellar lanaguage which introduces the following new reserved keywords and symbols:

```
match, default, {, }, ‘=>’
```

You must modify any Stellar expressions that use these keywords not in quotes.

4. Troubleshooting

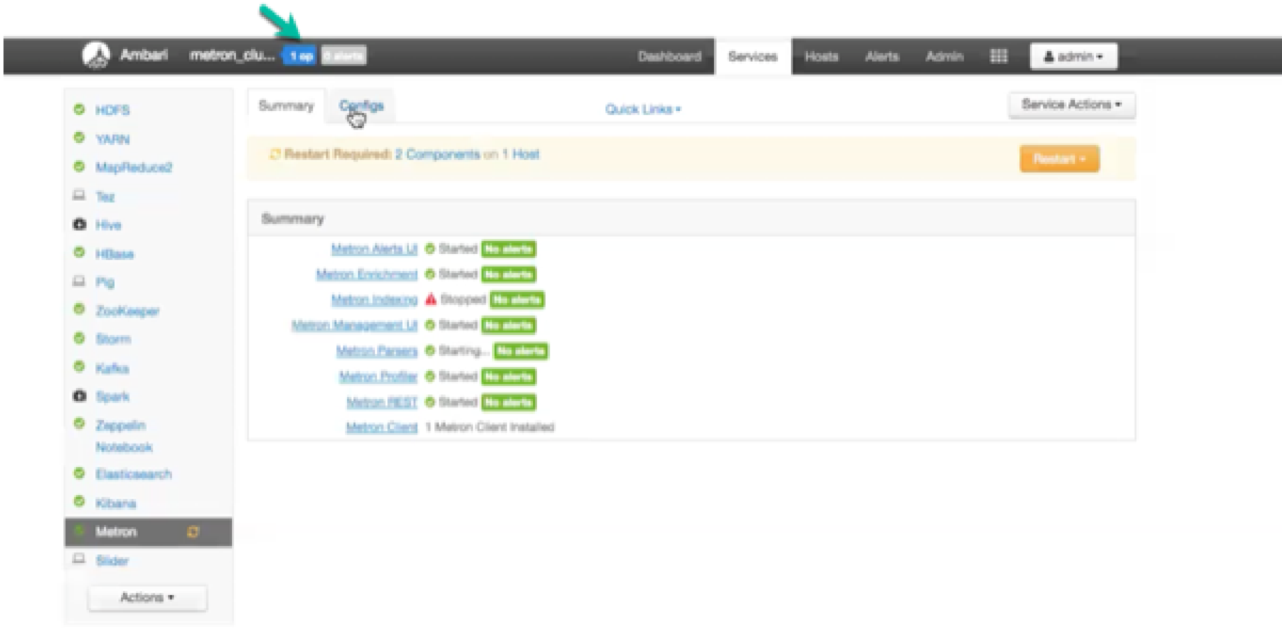
If you run into issues with your upgrade use the following troubleshooting tips to identify and resolve those issues.

4.1. Checking the Status of the Parsers

If your parsers do not restart, you can check the status of the parsers by completing the following steps:

1. Click the operation status tab at the top of the Ambari window.

Figure 4.1. Ambari Summary Tab



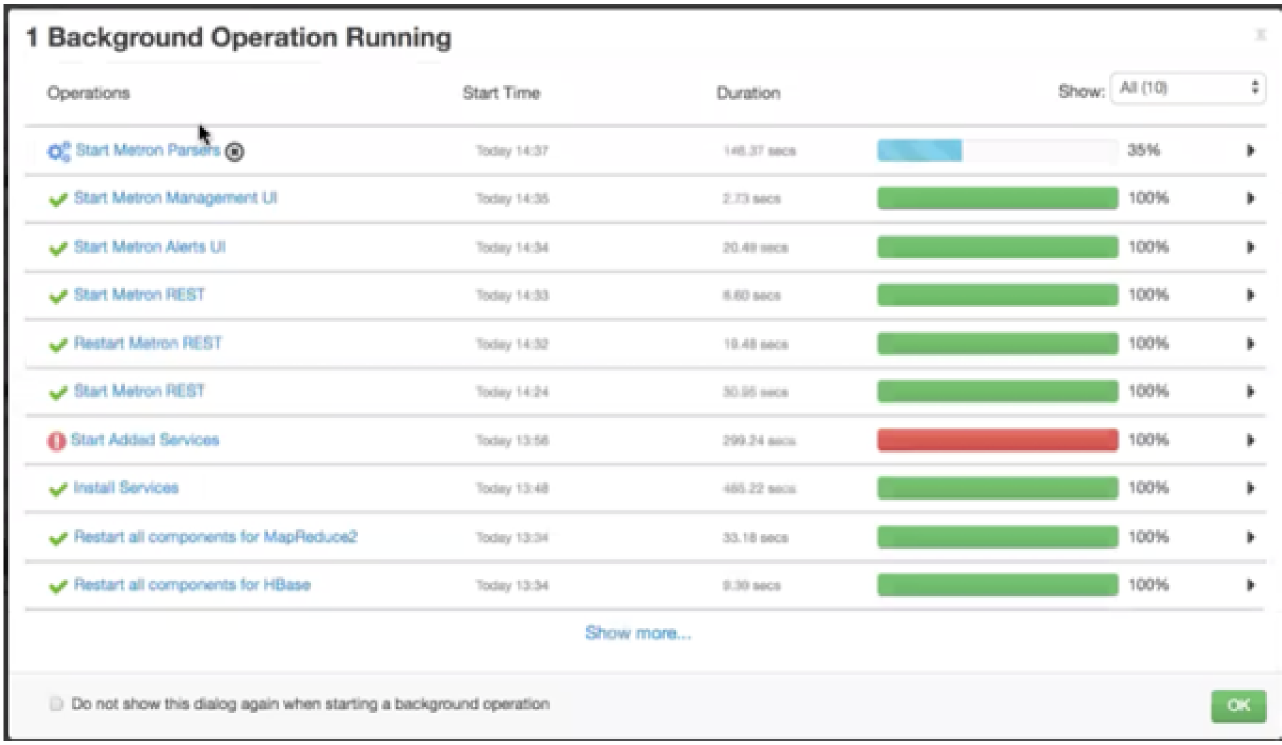
Restart Required: 2 Components on 1 Host

Summary

Metron Alerts UI	Started	No alerts
Metron Enrichment	Started	No alerts
Metron Indexing	Stopped	No alerts
Metron Management UI	Started	No alerts
Metron Parsers	Starting...	No alerts
Metron Profiler	Started	No alerts
Metron REST	Started	No alerts
Metron Client	1 Metron Client Installed	

Ambari displays the Operations Running Status window.

Figure 4.2. Ambari Background Operation Page



1 Background Operation Running

Show: All (10)

Operations	Start Time	Duration		
Start Metron Parsers	Today 14:37	146.37 secs	35%	▸
✓ Start Metron Management UI	Today 14:35	2.73 secs	100%	▸
✓ Start Metron Alerts UI	Today 14:34	20.49 secs	100%	▸
✓ Start Metron REST	Today 14:33	8.60 secs	100%	▸
✓ Restart Metron REST	Today 14:32	19.48 secs	100%	▸
✓ Start Metron REST	Today 14:24	30.95 secs	100%	▸
! Start Added Services	Today 13:56	299.24 secs	100%	▸
✓ Install Services	Today 13:48	465.22 secs	100%	▸
✓ Restart all components for MapReduce2	Today 13:34	33.18 secs	100%	▸
✓ Restart all components for HBase	Today 13:34	9.39 secs	100%	▸

Show more...

□ Do not show this dialog again when starting a background operation

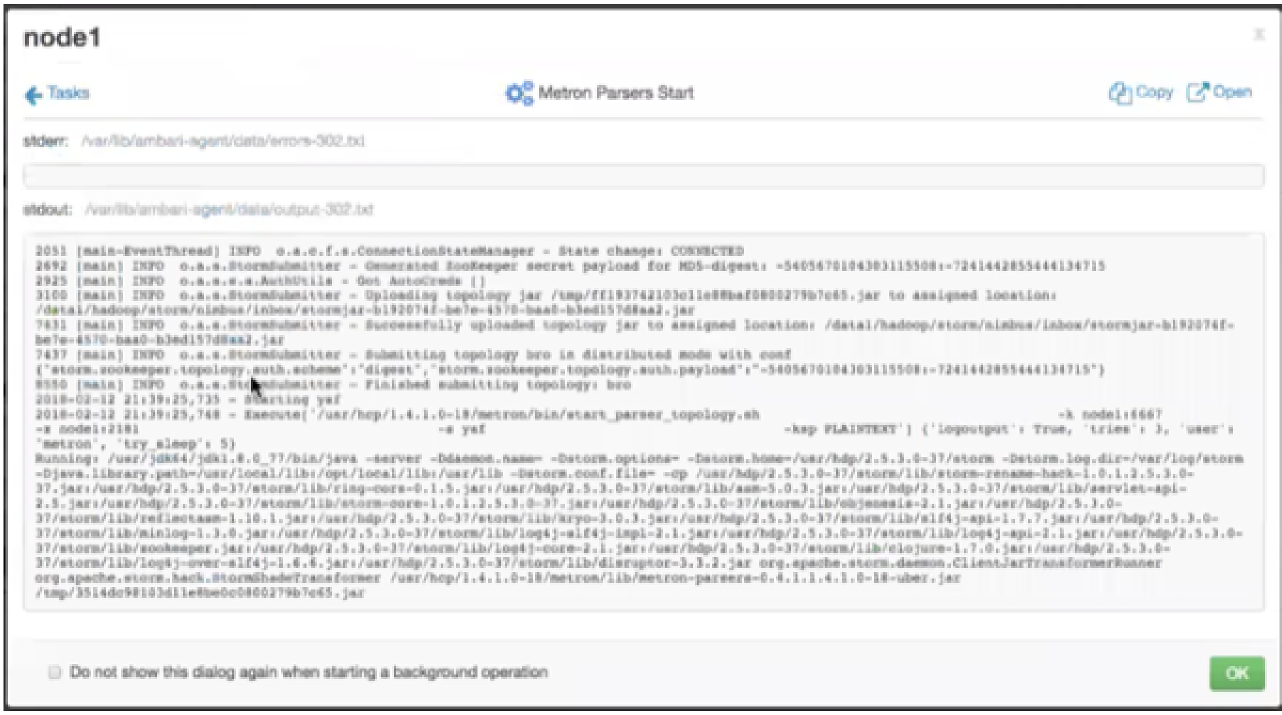
2. Click **Start Metron Parsers**.

   Ambari displays the **Start Metron Parsers** window.

3. Click the parser node you want to check, then click **Metron Parsers Start**.

   Ambari displays information on the status of the parser.

Figure 4.3. Metron Parsers Start Page



stderr: /var/lib/ambari-agent/data/errors-302.txt

stdout: /var/lib/ambari-agent/data/output-302.txt

```
2051 [main-EventThread] INFO  o.a.c.f.s.ConnectionStateManager - State change: CONNECTED
2692 [main] INFO  o.a.s.StormSubmitter - Generated ZooKeeper secret payload for MD5-digest: -5405670104303115508:-7241442855444134715
2925 [main] INFO  o.a.s.s.a.AuthUtils - Got AutoCreds []
3100 [main] INFO  o.a.s.StormSubmitter - Uploading topology jar /tmp/ff193742103c11e88baf0800279b7c65.jar to assigned location:
/data1/hadoop/storm/nimbus/inbox/stormjar-b192074f-be7e-4570-baa0-b3ed157d8aa2.jar
7431 [main] INFO  o.a.s.StormSubmitter - Successfully uploaded topology jar to assigned location: /data1/hadoop/storm/nimbus/inbox/stormjar-b192074f-
be7e-4570-baa0-b3ed157d8aa2.jar
7437 [main] INFO  o.a.s.StormSubmitter - Submitting topology bro in distributed mode with conf
{"storm.zookeeper.topology.auth.scheme":"digest","storm.zookeeper.topology.auth.payload":"-5405670104303115508:-7241442855444134715"}
8550 [main] INFO  o.a.s.StormSubmitter - Finished submitting topology: bro
2018-02-12 21:39:25,735 - Starting yaf
2018-02-12 21:39:25,748 - Execute['/usr/hcp/1.4.1.0-18/metron/bin/start_parser_topology.sh                                   -k node1:6667
-z node1:2181                                          -s yaf                                           -ksp PLAINTEXT'] {'logoutput': True, 'tries': 3, 'user':
'metron', 'try_sleep': 5}
Running: /usr/jdk64/jdk1.8.0_77/bin/java -server -Ddaemon.name= -Dstorm.options= -Dstorm.home=/usr/hdp/2.5.3.0-37/storm -Dstorm.log.dir=/var/log/storm
-Djava.library.path=/usr/local/lib:/opt/local/lib:/usr/lib -Dstorm.conf.file= -cp /usr/hdp/2.5.3.0-37/storm/lib/storm-rename-hack-1.0.1.2.5.3.0-
37.jar:/usr/hdp/2.5.3.0-37/storm/lib/ring-cors-0.1.5.jar:/usr/hdp/2.5.3.0-37/storm/lib/asm-5.0.3.jar:/usr/hdp/2.5.3.0-37/storm/lib/servlet-api-
2.5.jar:/usr/hdp/2.5.3.0-37/storm/lib/storm-core-1.0.1.2.5.3.0-37.jar:/usr/hdp/2.5.3.0-37/storm/lib/objenesis-2.1.jar:/usr/hdp/2.5.3.0-
37/storm/lib/reflectasm-1.10.1.jar:/usr/hdp/2.5.3.0-37/storm/lib/kryo-3.0.3.jar:/usr/hdp/2.5.3.0-37/storm/lib/slf4j-api-1.7.7.jar:/usr/hdp/2.5.3.0-
37/storm/lib/minlog-1.3.0.jar:/usr/hdp/2.5.3.0-37/storm/lib/log4j-slf4j-impl-2.1.jar:/usr/hdp/2.5.3.0-37/storm/lib/log4j-api-2.1.jar:/usr/hdp/2.5.3.0-
37/storm/lib/zookeeper.jar:/usr/hdp/2.5.3.0-37/storm/lib/log4j-core-2.1.jar:/usr/hdp/2.5.3.0-37/storm/lib/clojure-1.7.0.jar:/usr/hdp/2.5.3.0-
37/storm/lib/log4j-over-slf4j-1.6.6.jar:/usr/hdp/2.5.3.0-37/storm/lib/disruptor-3.3.2.jar org.apache.storm.daemon.ClientJarTransformerRunner
org.apache.storm.hack.StormShadeTransformer /usr/hcp/1.4.1.0-18/metron/lib/metron-parsers-0.4.1.1.4.1.0-18-uber.jar
/tmp/3514dc98103d11e8be0c0800279b7c65.jar
```

□ Do not show this dialog again when starting a background operation

4. Review the information in this window to determine the status of your parsers.