CLOUDERA

WHITE PAPER

OPTIMIZING SPLUNK LOG INGESTION WITH CLOUDERA DATAFLOW

DATAFLOW

Government agencies and commercial entities must retain data for several years and commonly experience IT challenges due to increased data volumes and new sources coming online. Due to these factors, they are starting to undergo degradation in the performance of Security Information & Event Management (SIEM's) tools like Splunk. To continue to meet mission needs, address the increase in data sources that require protection, and manage costs, they have started research strategies that complement their Splunk investments while looking for solutions that meet or exceed their organization's policies.

The following requirements must be met to increase the performance of Splunk and maximize IT investment:

- Route data to utilize cost-effective data destinations
- Preprocess incoming data by filtering and removing extraneous fields
- Aggregate logs
- Replay any data
- Easily move workloads to Splunk Cloud
- Standardize the onboarding of additional data sources
- Provide observability of data pipelines

This white paper will focus on how agencies use DataFlow for universal data distribution as a solution for Splunk optimization with the technical details required to re-create this work. This white paper does not provide details on how to deploy CDF, instead it includes sample data files and a template for a CDF data flow, that can be found here: https://github.com/Brookslan/UDDS-SIEM-Optimization

Also included is a data flow template demonstrating how to route, filter, and aggregate data from Windows Application Event or IP Stream logs. This approach is designed to be hybrid in nature, meaning it can be used on-premises, in the cloud, or as a combination of both. Due to the flexibility of this architecture, it can be slowly phased into an existing Splunk deployment without interrupting the service.

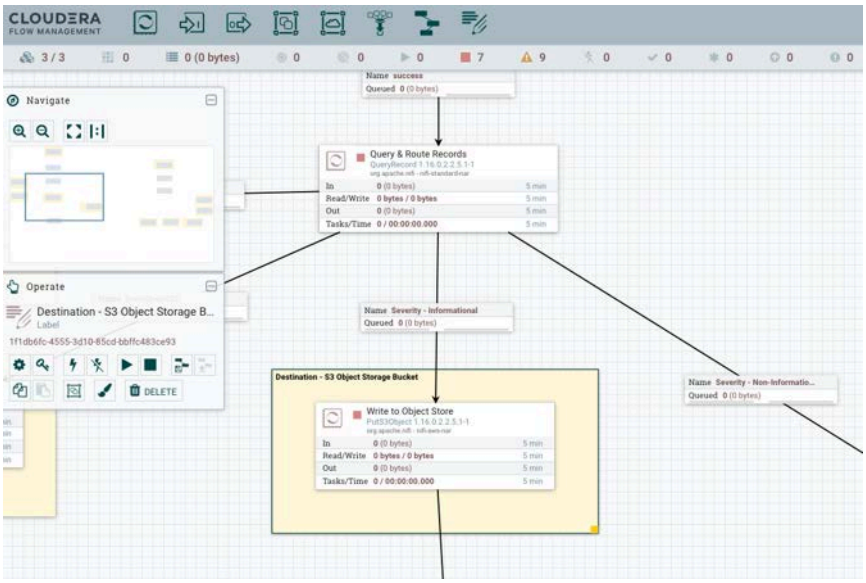
Routing Data To Utilize Cost-Effective Data Destinations

There are many reasons agencies are seeking to leverage more cost-effective data storage locations to continue to scale and meet business SLAs. They also need flexibility in terms of supporting future architectures and applications. Currently, object storage has become the industry standard due to its ability to:

- Grow to their needs as data volumes increase
- Continue to perform analytics at scale
- Allow for organizations to leverage data tiers (hot, medium, or cold) for storage optimization
- Facilitate a hybrid architecture

There are many different software and hardware vendors that support their object storage products, and many of them provide an API that is compliant with the AWS S3 API. For this white paper, we will be using the native capabilities of DataFlow to demonstrate writing a file as an object into a specific bucket location. To complete this task, the NiF processor PutS3Object is required.

In this image, we used the QueryRecords NiFi processor to query and route the corresponding results to a file, which can be set to a different file format (CSV, JSON). Once the output file has been created, we can write this to a pre-existing object storage bucket. This bucket can be hosted on-premise or in AWS cloud infrastructure.



In this set of images, we are showing the configuration values of the PutS3Object NiFi processor to communicate with the object storage instance. This example is using an on-premise deployment of object storage using Apache Ozone. The IP address host containing Ozone Manager is required to complete this. The user will also need to create a bucket in this object storage instance.

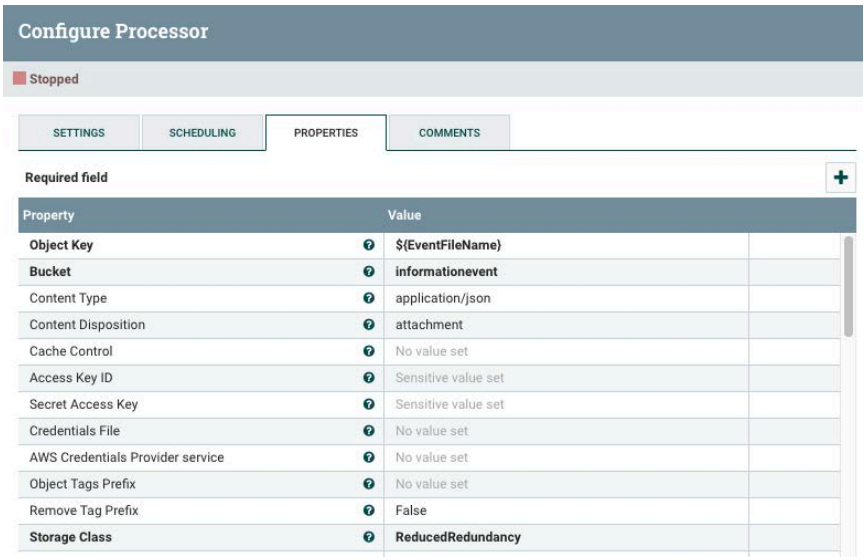
If you're using S3 in AWS's cloud, you will need to provide the proper Region of the bucket, and you can leave the Endpoint Override URL value empty. In your environment, the user list may need to be set from your environment, but we are using the user "hadoop" for these values.

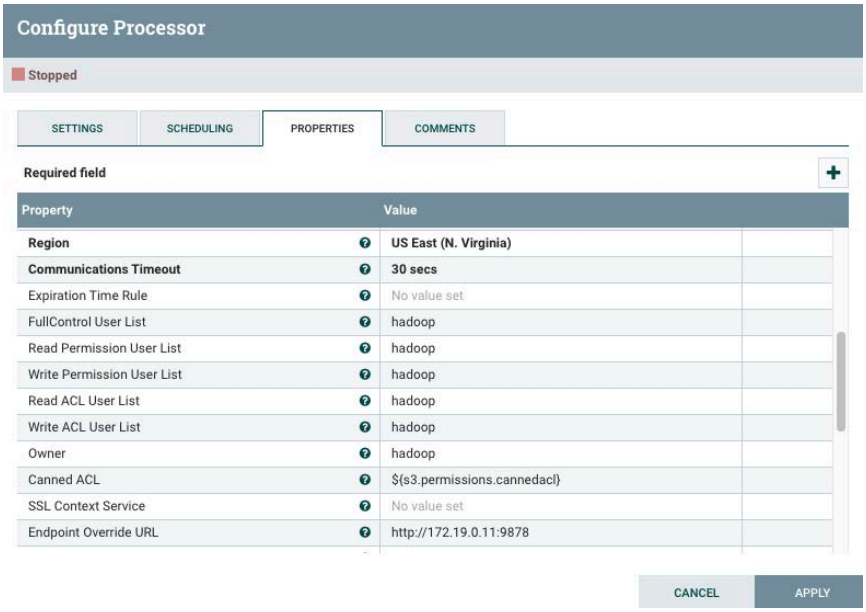
Please note that we have security enforcement on the AWS S3 protocol turned off. If security enforcement is turned on, the user must provide the values for Access Key ID and Secret Access Key that match their AWS credentials.

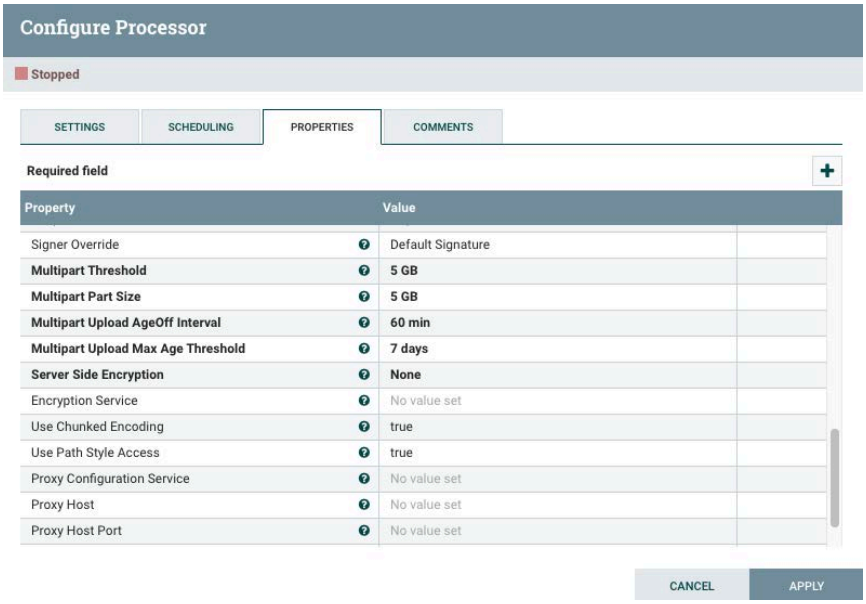
The configuration fields that are important to enter correctly are:

The configuration fields that are important to enter correctly are:

- Object Key: **${filename}**
- Bucket: **bucket1**
- Storage Class: **ReducedRedundancy**
- Endpoint Override URL: **http://<Object Store Host IP>:<Object Store Port>**
- Use Path Style Access: **true**
- Access Key: **testUser**
- Secret Access Key: **123456**




Configure Processor
Stopped
SETTINGS
SCHEDULING
PROPERTIES
COMMENTS
Required field
Property
Value
Region
US East (N. Virginia)
Communications Timeout
30 secs
Expiration Time Rule
No value set
FullControl User List
hadoop
Read Permission User List
hadoop
Write Permission User List
hadoop
Read ACL User List
hadoop
Write ACL User List
hadoop
Owner
hadoop
Canned ACL
${s3.permissions.cannedacl}
SSL Context Service
No value set
Endpoint Override URL
http://172.19.0.11:9878
CANCEL
APPLY


Configure Processor
Stopped
SETTINGS
SCHEDULING
PROPERTIES
COMMENTS
Required field
Property
Value
Signer Override
Default Signature
Multipart Threshold
5 GB
Multipart Part Size
5 GB
Multipart Upload AgeOff Interval
60 min
Multipart Upload Max Age Threshold
7 days
Server Side Encryption
None
Encryption Service
No value set
Use Chunked Encoding
true
Use Path Style Access
true
Proxy Configuration Service
No value set
Proxy Host
No value set
Proxy Host Port
No value set
CANCEL
APPLY

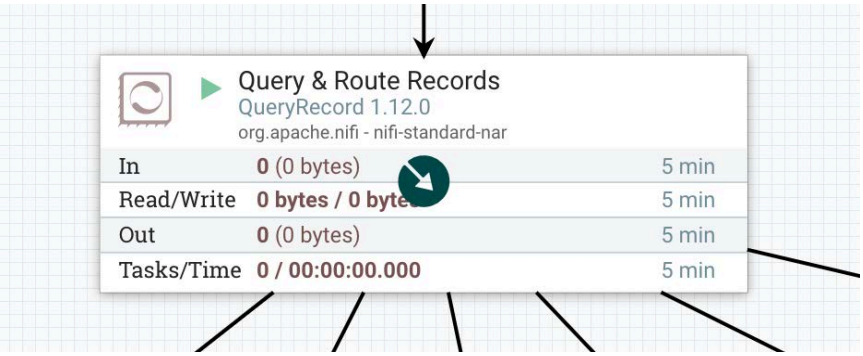
Pre-Process Incoming Data by Filtering and Removing Extraneous Fields

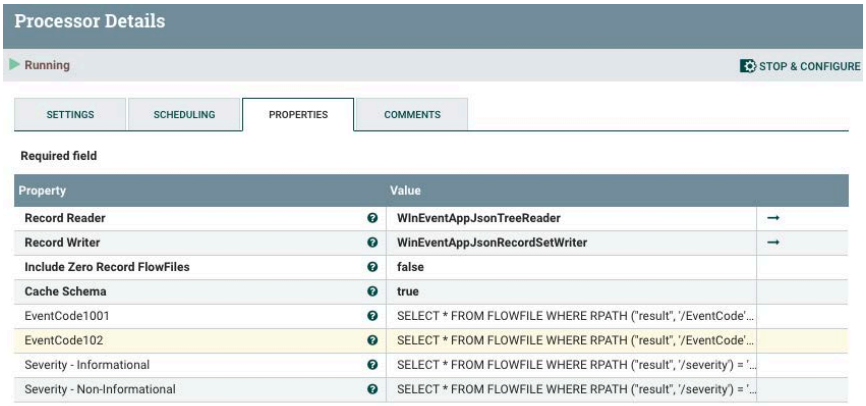
To improve the performance of Splunk or other SIEMs, it is critical to be selective about the data sent to be indexed and searched in the future. There are many advantages to limiting the amount of data an analyst has to sift through later. With less data to be searched through, it allows for queries to be completed quicker, and critical events can be given more attention or immediate action.

Concerning filtering data, CDF can complete this task using native processors. Data can also be filtered based on any data or metadata associated with the flow file. In the following example, we will be using the event severity level as a filter to determine if data should be routed to our SIEM or object storage. This example will pull the severity level, which is a metadata value, and only send events that are non-information to our SIEM (i.e., Splunk). Information events will be routed to our object storage bucket.

To complete this task, we will use the QueryRecord NiFi processor to query incoming records for severity level and event codes. This acts as a filter and routes the results based on the query. In this example, the following SQL statements are used for severity and event codes. Please note, the RPATH call is used to traverse through nested JSON elements in the input file.

- Severity - Information
  SELECT * FROM FLOWFILE WHERE RPATH ("result", '/severity') = 'Information'
- Severity - Warning
  SELECT * FROM FLOWFILE WHERE RPATH ("result", '/severity') = 'Warning'
- Event Code 102
  SELECT * FROM FLOWFILE WHERE RPATH ("result", '/EventCode') = '102'
- Event Code 1001
  SELECT * FROM FLOWFILE WHERE RPATH ("result", '/EventCode') = '1001'

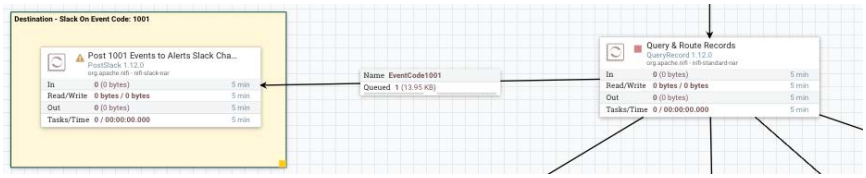




Removing unrequired or extra fields can be accomplished using the built-in SQL capabilities in the QueryRecords NiFi processor. Once those attributes are present in the query, they will be added to the output file that would be sent to a SIEM or an object storage instance.

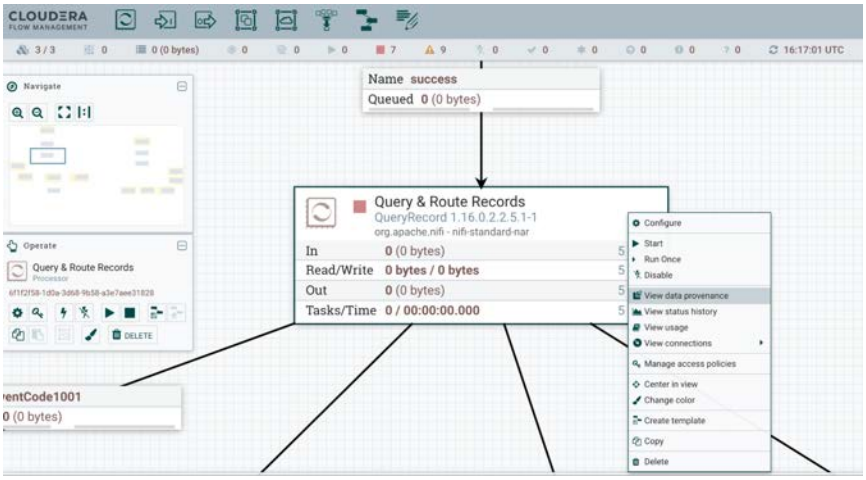
Aggregate Logs

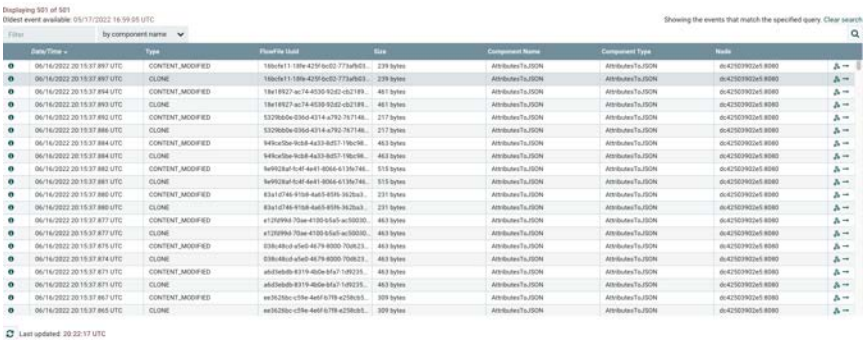
Log aggregation can enable organizations with greater control over the flow of data through their infrastructure as well as control over how data is written to its destination. In many ways, this can be considered transformation in an ETL process. By taking advantage of the native capabilities of the QueryRecords NiFi processor, the results from the queries are merged together out of the box. This allows developers to leverage all of the results of the query into a single output file. The following images show the results of the query merged into a single output file.

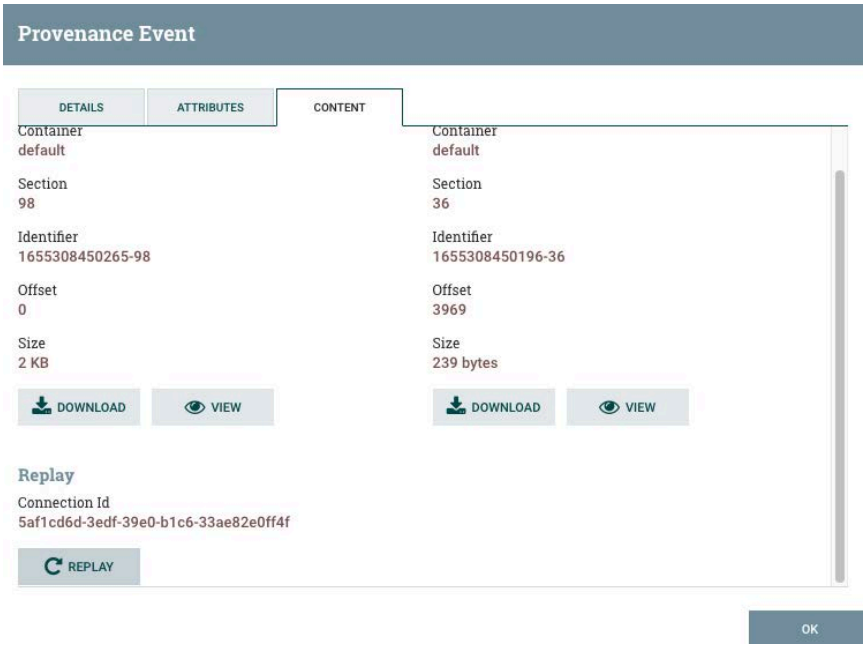


Replay Any Data

All NiFi processors can replay any data that has been processed through them. This capability is a valuable tool for development or debugging purposes and is very easy to use. In the following images, a user must right-click on the processor of interest and select the option to view data provenance. A list of events will be displayed to the user, and one of those events can be selected. Under the Content tab, the replay button is available, which can be clicked to replay that data.

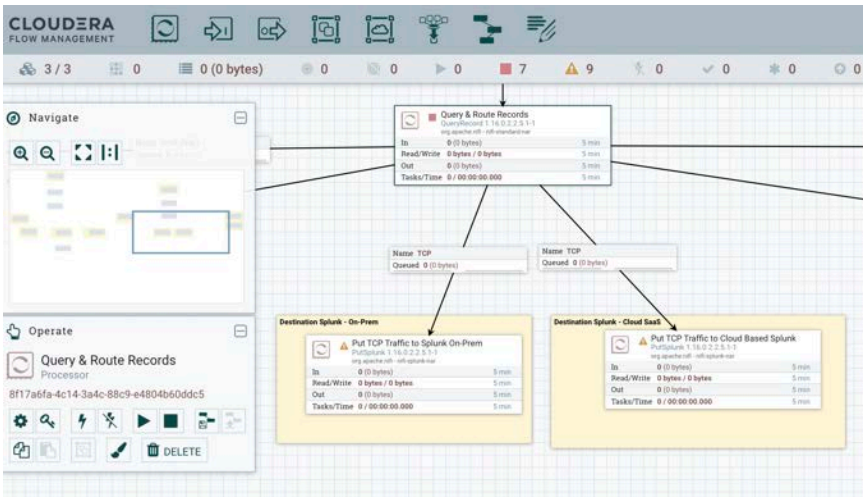

CLOUDERA
FLOW MANAGEMENT
Name success
Queued 0 (0 bytes)
Query & Route Records
QueryRecord 1.16.0.2.2.5.1-1
org.apache.nifi - nifi-standard-nar
In 0 (0 bytes)
Read/Write 0 bytes / 0 bytes
Out 0 (0 bytes)
Tasks/Time 0 / 00:00:00.000
Configure
Start
Run Once
Disable
View data provenance
View status history
View usage
View connections
Manage access policies
Center in view
Change color
Create template
Copy
Delete
Navigate
Operate
Query & Route Records



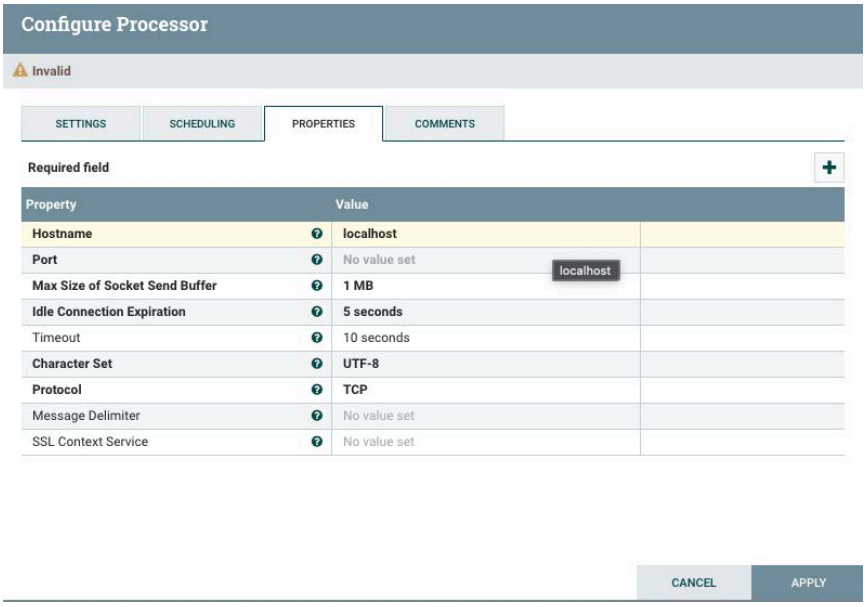

Provenance Event
DETAILS
ATTRIBUTES
CONTENT
Container
default
Section
98
Identifier
1655308450265-98
Offset
0
Size
2 KB
DOWNLOAD
VIEW
Container
default
Section
36
Identifier
1655308450196-36
Offset
3969
Size
239 bytes
DOWNLOAD
VIEW
Replay
Connection Id
5af1cd6d-3edf-39e0-b1c6-33ae82e0ff4f
REPLAY
OK

Easily Move Workloads to Splunk Cloud with Built-In Connectors

Moving data into a cloud-based Splunk instance can be accomplished using the native NiFi built-in processor called PutSplunk. This processor easily allows organizations to push data from their on-premise or cloud environment to Splunk instances regardless of location. By taking advantage of this capability, SIEM workloads can be straightforwardly moved to the cloud in parallel as depicted below.



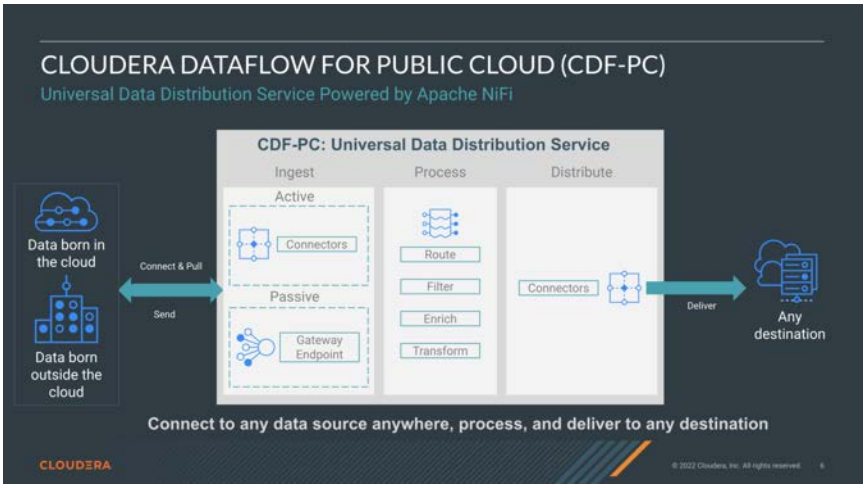
The configuration of this processor must be set with the hostname of the Splunk instance. The following images display the processor and configuration screen. For this white paper, the specific Splunk values have not been included.



Standardize the Onboarding of Additional Data Sources

One of the significant advantages of using CDF as a UDDS is the ability to standardize how new data sources get collected and moved throughout an enterprise - build once, use many times.

By giving teams a standard approach to ingesting new data sources, organizations can streamline their process to acquire new datasets and use data in their missions or feed downstream applications. This approach decouples data sources from their destinations and brings immediate business value.

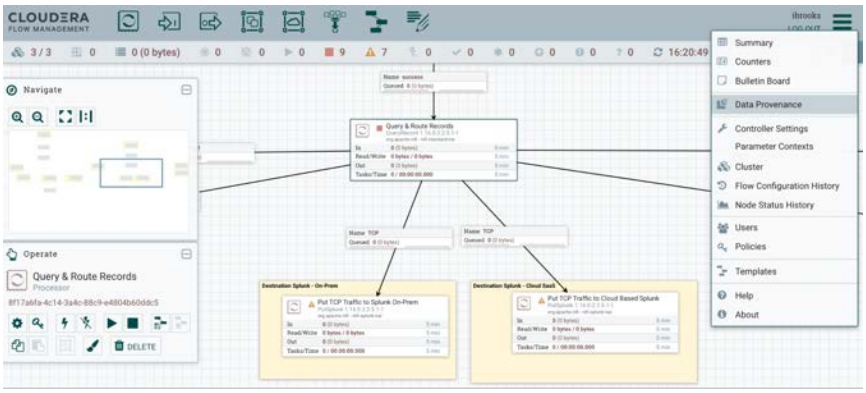


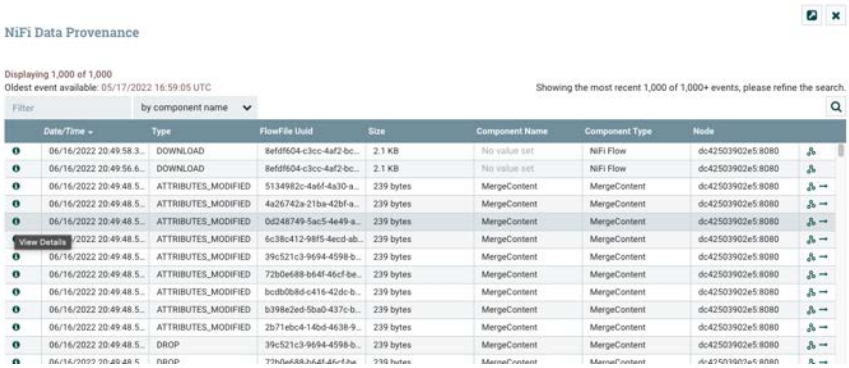
Provide Observability of Data Pipelines

Organizations require observability of their data pipelines. Observability is critical in validating chains of custody for audits and validation, but it's also essential for performance. CDF keeps a very granular level of detail about each piece of data that it ingests. According to the Apache NiFi documentation, "As the data is processed hrough the system, transformed, routed, split, aggregated, and distributed to other endpoints, an audit trail is created and stored within Apache NiFi's Provenance Repository." This implies that any and all steps that are used to process data can be stored and tracked.

We can select Data Provenance from the Global Menu. All provenance events will be listed, and the complete data provenance can be viewed by selecting the icon on the right-hand side of the event. The out-of-the-box observability capabilities of CDF allow for a comprehensive view of data pipelines, which is critical for organizations today.

The following images detail how the complete data provenance listing can be selected for all events. The last image displays all of the steps used for a particular event from the moment the data was acquired and ingested into CDF.





About Cloudera

At Cloudera, we believe that data can make what is impossible today, possible tomorrow. We empower people to transform complex data into clear and actionable insights. Cloudera delivers an enterprise data cloud for any data, anywhere, from the Edge to AI. Powered by the relentless innovation of the open source community, Cloudera advances digital transformation for the world's largest enterprises.

Learn more at cloudera.com

Connect with Cloudera

About Cloudera:
cloudera.com/more/about.html

Read our blog:
blog.cloudera.com

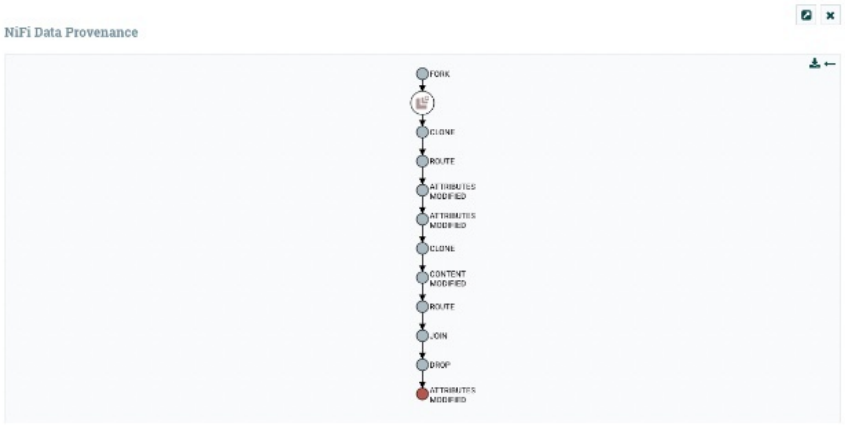
Follow us on Twitter:
twitter.com/cloudera

Visit us on Facebook:
facebook.com/cloudera

See us on YouTube:
youtube.com/cloudera

Join the Cloudera Community:
community.cloudera.com

Read about our customers' successes:
cloudera.com/more/customers



Conclusion

Government agencies and commercial entities will need to continue to address growing requirements and IT challenges. They will need solutions that are flexible enough to enable hybrid deployments, possess the ability to scale to the growing volume of data, and complement existing IT investments like Splunk. In addition to these IT challenges, they will need an universal solution that can ingest data from new sources as they come online, such as IoT devices, and deliver data to destinations, such as cloud based applications or future storage devices. Due to these factors, using CDF as an UDDS allows organizations to address the degradation in the performance of their SIEMs, and allow them to continue to meet and exceed the future needs of the mission.

Get Started Today

Wherever you are on your hybrid cloud journey, a first-class data distribution service is critical for successfully adopting a modern hybrid data stack. Cloudera DataFlow for the Public Cloud (CDF-PC) provides a universal, hybrid, and streaming-first data distribution service that enables customers to gain control of their data flows.

Take our interactive product tour:
https://www.cloudera.com/products/dataflow/cdp-tour-dataflow.html

Sign up for a free trial: https://www.cloudera.com/campaign/try-cdp-public-cloud.html

Cloudera, Inc. 5470 Great America Pkwy, Santa Clara, CA 95054 USA cloudera.com